

# CLASSE ARRAYS



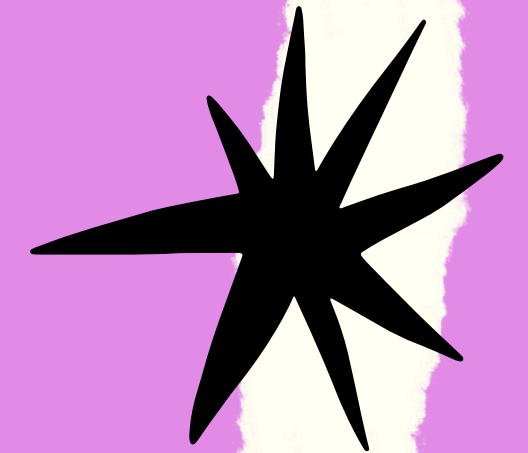
```
import java.util.Arrays;
public class ClasseArray {
    public static void main(String[] args) {

        double[] vetorDouble = { 8.4, 9.3, 0.2, 7.9, 3.4 };
        Arrays.sort(vetorDouble); // classifica em ordem crescente
        System.out.print("\nVETOR ORDENADO: ");
        //instrução "for" aprimorada - (estilo for .. each)
        for (double valor : vetorDouble) {
            System.out.printf("%.1f - ", valor);
        }

        int[] vetorPreenchido = new int[10];
        Arrays.fill(vetorPreenchido, 7); // preenche o vetor com 7s
        mostraVetor(vetorPreenchido, "VETOR PREENCHIDO");

        int[] vetorInt = { 1, 2, 3, 4, 5, 6, 7 };
        int[] copiaVetorInt = new int[vetorInt.length];
        // copia vetorInt em copiaVetorInt
        //(origem, posição inicial, destino, posição inicial, quantos elementos)
        System.arraycopy(vetorInt, 3, copiaVetorInt, 0, vetorInt.length - 3);
        mostraVetor(vetorInt, "VETOR INTEIRO");
        mostraVetor(copiaVetorInt, "COPIA DO VETOR");

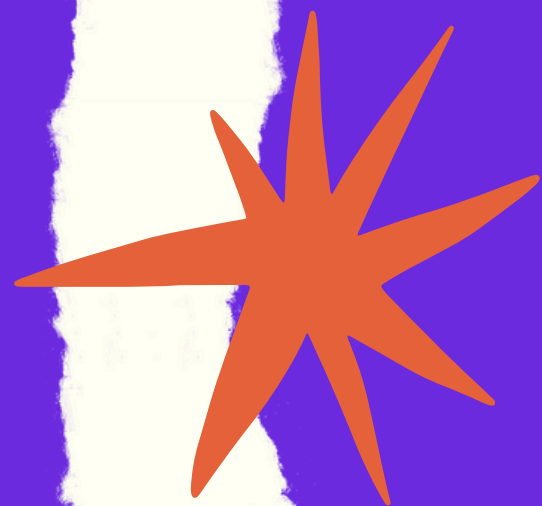
        // verifica a igualdade entre dois vetores
        boolean b = Arrays.equals(vetorInt, copiaVetorInt);
        // if ternário: condição ? verdadeiro : falso
        System.out.printf("\nvetorInt %s copiaVetorInt\n", (b ? "==" : "!="))
    }
}
```



```
// pesquisa o valor 5 em vetorInt
int local = Arrays.binarySearch(vetorInt, 5);
// if ternário
String msg = local >= 0 ?
    "\nEncontrado 5 na posicao " + local :
    "\n5 nao encontrado no vetor ";
System.out.println(msg);
//usa o método procuraVetor
procuraVetor(vetorInt, 42);
}
```

```
//gera saída de dados para vetor inteiro
public static void mostraVetor(int[] vetor, String descricao) {
    System.out.printf("\n%s: ", descricao);
    for (int valor : vetor) {
        System.out.printf("%d - ", valor);
    }
}
```

```
public static void procuraVetor(int[] vetor, int valor) {
    int local = Arrays.binarySearch(vetor, valor);
    String msg = local >= 0 ?
        ("\nEncontrado " + valor + " na posicao " + local) :
        ("\n" + valor + " nao encontrado no vetor ");
    System.out.println(msg);
}
```

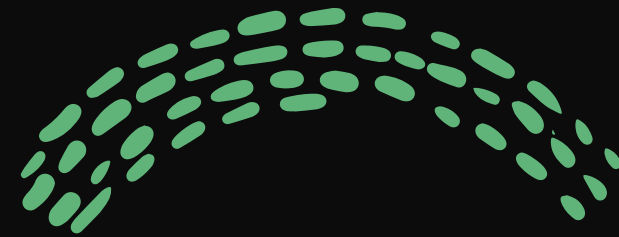




```
VETOR ORDENADO: 0,2 - 3,4 - 7,9 - 8,4 - 9,3 -  
VETOR PREENCHIDO: 7 - 7 - 7 - 7 - 7 - 7 - 7 - 7 - 7 - 7 - 7  
VETOR INTEIRO: 1 - 2 - 3 - 4 - 5 - 6 - 7 -  
COPIA DO VETOR: 4 - 5 - 6 - 7 - 0 - 0 - 0 -
```

```
vetorInt != copiaVetorInt
```

```
Encontrado 5 na posicao 4
```



```
42 nao encontrado no vetor
```





# CLASSE STRING



```
String s0 = "como vai";
char[] vetorChar = new char[5];
System.out.printf("%ns0: %s", s0);

//tamanho da string: length
System.out.printf("%nTamanho de s0: %d", s0.length());
System.out.printf("%nORDEM REVERSA: ");
for (int cont = s0.length() - 1; cont >= 0; cont--) {
    System.out.printf("%c ", s0.charAt(cont));
}

// copia caracteres de string para vetor de char: getChars
s0.getChars(0, 5, vetorChar, 0);
System.out.printf("%nVETOR DE CARACTERES: ");
for (char caractere : vetorChar) {
    System.out.print(caractere);
}
System.out.println();
```



```
s0: como vai
Tamanho de s0: 8
ORDEM REVERSA: i a v o m o c
VETOR DE CARACTERES: como
```

```
// comparação de strings
String s1 = new String("Que o dia"), s2 = "seja feliz";
String s3 = "Feliz Aniversario", s4 = "feliz aniversario";

// usando equals (testa quanto à igualdade)
if (s1.equals("Que o dia"))
    System.out.println("s1 igual a \"Que o dia\" ");

// usando equals (ignorando maiúsculas e minúsculas)
if (s3.equalsIgnoreCase(s4))
    System.out.printf("%s igual a %s %n", s3, s4);

// usando o == (compara objetos)
if (s1 == "Que o dia")
    System.out.println("s1 eh o mesmo objeto que \"Que o dia\" ");
else
    System.out.println("s1 nao eh o mesmo objeto que \"Que o dia\" ");
```



```
s1 igual a "Que o dia"
Feliz Aniversario igual a feliz aniversario
s1 nao eh o mesmo objeto que "Que o dia"
```



```
String s1 = new String("Que o dia"), s2 = "seja feliz";  
String s3 = "Feliz Aniversario", s4 = "feliz aniversario";  
  
//usando o compareTo  
System.out.printf("%n s1.compareTo(s2)= %d", s1.compareTo(s2));  
System.out.printf("%n s2.compareTo(s2)= %d", s2.compareTo(s1));  
System.out.printf("%n s1.compareTo(s1)= %d", s1.compareTo(s1));  
System.out.printf("%n s4.compareTo(s3)= %d", s4.compareTo(s3));  
System.out.printf("%n s3.compareTo(s4)= %d", s3.compareTo(s4));  
  
//usando regionMatches  
if (s3.regionMatches(0, s4, 0, 5))  
    System.out.println  
        ("\n5 primeiros caracteres de s3 e s4 coincidem");  
else  
    System.out.println  
        ("\n5 primeiros caracteres de s3 e s4 nao coincidem");  
if (s3.regionMatches(true, 0, s2, 5, 5)) // sem case sensitive  
    System.out.println  
        ("\n5 primeiros caracteres de s3 coincidem com os de s2");
```



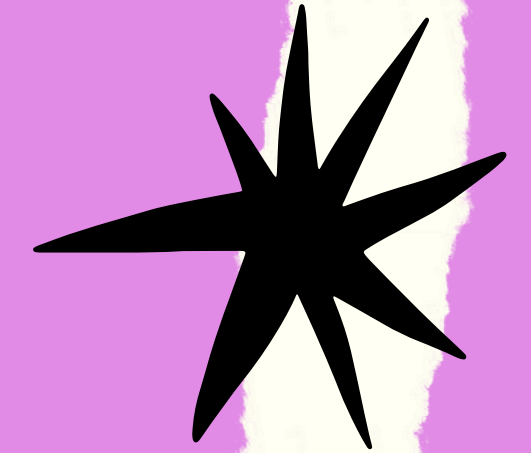
```
s1.compareTo(s2)= -34  
s2.compareTo(s2)= 34  
s1.compareTo(s1)= 0  
s4.compareTo(s3)= 32  
s3.compareTo(s4)= -32  
Os 5 primeiros caracteres de s3 e s4 nao coincidem  
Os 5 primeiros caracteres de s3 coincidem com os de s2
```



```
//métodos startswith e endswith
String[] strings = {"inicio", "iniciando", "final", "finalizando"};
for (String string : strings) {
    if (string.startsWith("in"))
        System.out.printf("\'%s\' começa com \'in\' %n", string);

    if (string.startsWith("ci", 3)) //índice que a comparação inicia
        System.out.printf
            ("\'%s\' começa com \'ci\' na posicao 4%n", string);

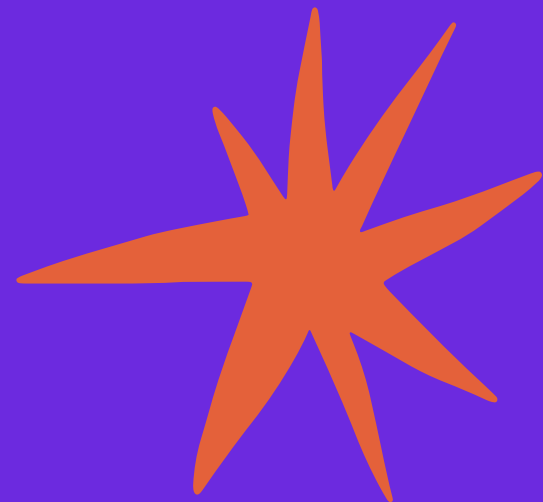
    if (string.endsWith("ando"))
        System.out.printf
            ("\'%s\' termina com \'ando\' %n", string);
}
```



```
"inicio" começa com "in"
"inicio" começa com "ci" na posicao 4
"iniciando" começa com "in"
"iniciando" começa com "ci" na posicao 4
"iniciando" termina com "ando"
"finalizando" termina com "ando"
```



```
//métodos de pesquisa
String letras = "abcdefghijklmabcdefghijklm";
System.out.printf
    ("'k' está localizado em %d %n", letras.indexOf('k'));
System.out.printf
    ("'a' está localizado em %d %n", letras.indexOf('a', 1));
System.out.printf
    ("'$' está localizado em %d %n", letras.indexOf('$'));
System.out.printf
    ("'def' está localizado em %d %n", letras.indexOf("def"));
System.out.printf
    ("o ultimo 'k' está localizado em %d %n", letras.lastIndexOf('k'));
```



```
'k' está localizado em 10
'a' está localizado em 13
'$' está localizado em -1
'def' está localizado em 3
o ultimo 'k' está localizado em 23
```

```
//substring
System.out.printf
("Substring do 20 ate o final: %s %n", letras.substring(20));
System.out.printf
("A partir do 3 até o 6 (não incluso): %s %n", letras.substring(3, 6));

s1 = "feliz"; s2 = "ANIVERSARIO"; s3 = "      ESPACOS      ";
System.out.printf("s1 = %s%s2 = %s%s3 = %s%n", s1, s2, s3);

//replace
System.out.printf
("Troca o 'l' por 'L' em s1: %s%n", s1.replace('l', 'L'));
```



```
Substring do 20 ate o final: hijklm
A partir do 3 até o 6 (não incluso): def
s1 = feliz
s2 = ANIVERSARIO
s3 =      ESPACOS
Troca o 'l' por 'L' em s1: feLiz
```



```
//toLowerCase e toUpperCase
System.out.printf
("s1.toUpperCase(): %s\n", s1.toUpperCase());
System.out.printf
("s2.toLowerCase(): %s\n", s2.toLowerCase());

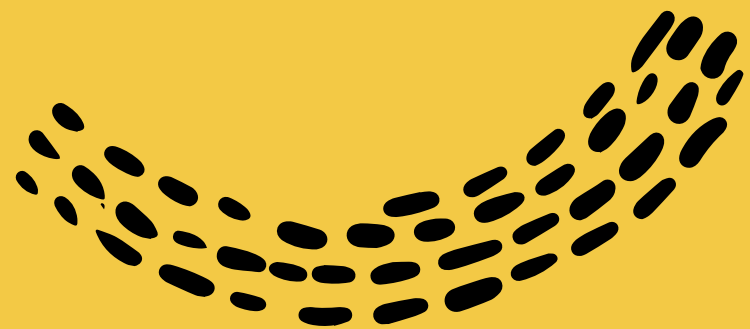
//trim: tira os espaços antes e depois
System.out.printf
("tam s3 antes do trim: %d e depois: %d\n", s3.length(), s3.trim().length());

//transforma string em vetor de char
vetorChar = s1.toCharArray();
for (char caractere : vetorChar)
    System.out.print(caractere + " - ");
System.out.println();
```

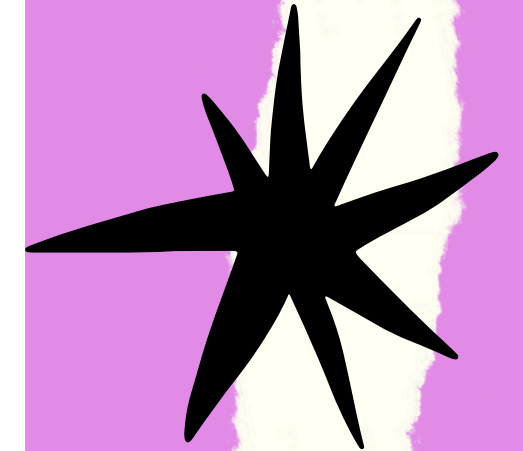


```
s1.toUpperCase(): FELIZ
s2.toLowerCase(): aniversario
tam s3 antes do trim: 17 e depois: 7
f - e - l - i - z -
```

# CLASSE ARRAYLIST

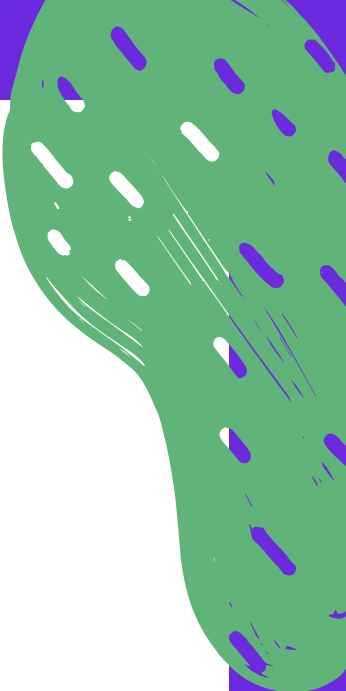
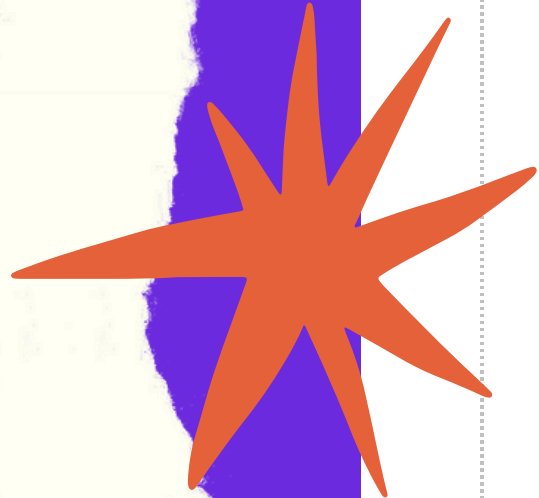
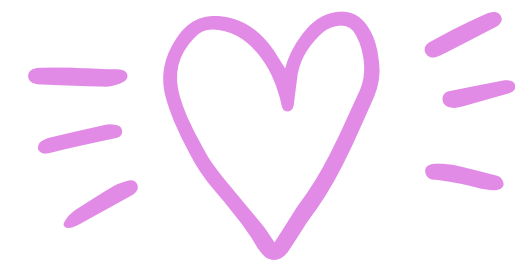


```
import java.util.ArrayList;
public class ClasseArrayList {
    public static void main(String[] args) {
        // por padrão, a quantidade de elementos é 10
        ArrayList<String> cores = new ArrayList<String>();
        ArrayList<Integer> num = new ArrayList<Integer>(7);
        cores.add("vermelho");
        display(cores, "CORES");
        cores.add(1, "amarelo");
        display(cores, "INSERIU O AMARELO NA POSIÇÃO 1");
        cores.add(0, "azul");
        display(cores, "INSERIU O AZUL NA POSIÇÃO 0");
        cores.add("verde"); // insere no final
        cores.add("branco");
        cores.add("preto");
        cores.add("laranja");
        cores.add("marron");
        cores.add("roxo");
        cores.add("rosa");
        cores.add("cinza"); //observe que ultrapassou 10 elementos
        cores.add("branco");
        System.out.println("LISTA DE CORES");
        for (int i = 0; i < cores.size(); i++) {
            System.out.printf("[%02d] %s\n", i + 1, cores.get(i));
        }
    }
}
```



```
System.out.println("LISTA DE CORES");  
for (int i = 0; i < cores.size(); i++) {  
    System.out.printf("[%02d] %s\n", i + 1, cores.get(i));  
}  
cores.remove("branco"); //remove o primeiro "branco"  
cores.remove(7); //remove o 6º elemento  
display(cores, "LISTA COM ITENS REMOVIDOS");  
//verifica se um valor está na lista  
System.out.printf("vermelho %s na lista\n",  
    cores.contains("vermelho") ? "esta" : "nao esta");  
System.out.printf("roxo %s na lista\n",  
    cores.contains("roxo") ? "esta" : "nao esta");  
//mostra o número de elementos da lista  
System.out.printf("Tamanho = %d\n", cores.size());  
// mostra o índice de determinado elemento  
System.out.printf("Índice do laranja: %d\n", cores.indexOf("laranja"))  
}
```

```
public static void display(ArrayList<String> items, String header) {  
    System.out.println(header);  
    for (String item : items) {  
        System.out.printf("  %s\n", item);  
    }  
    System.out.println();  
}
```



CORES

vermelho

INSERIU O AMARELO NA POSIÇÃO 1

vermelho amarelo

INSERIU O AZUL NA POSIÇÃO 0

azul vermelho amarelo

LISTA DE CORES

[01]azul

[02]vermelho

[03]amarelo

[04]verde

[05]branco

[06]preto

[07]laranja

[08]marron

[09]roxo

[10]rosa

[11]cinza

[12]branco



LISTA COM ITENS REMOVIDOS

azul

vermelho

amarelo

verde

preto

laranja

marron

rosa

cinza

branco

vermelho esta na lista

roxo nao esta na lista

Tamanho = 10

Indice do laranja: 5







**OBRIÇADA!**

Alguma Pergunta?

[professora@lucilia.com.br](mailto:professora@lucilia.com.br)

