



# CURSO DE PROGRAMAÇÃO EM JAVA



1.

# Arrays

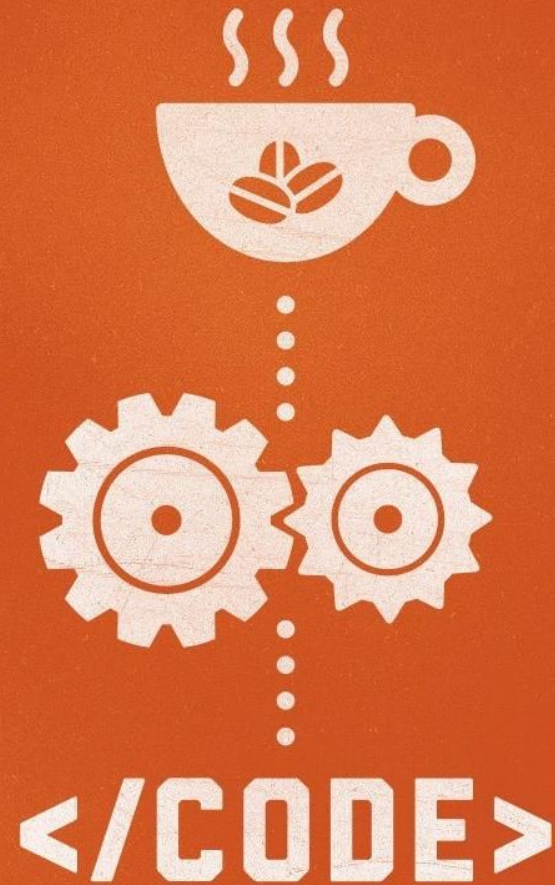
## Arrays

Um array é um grupo de variáveis (**elementos** ou **componentes**) que contém valores todos do mesmo tipo. Os arrays são objetos, portanto, considerados **tipos por referência**.

Os elementos do array podem ser **tipos primitivos** ou **tipos por referência** (inclusive arrays).

Para referenciar em elemento particular em um array, especificamos o nome da referência para o array e o número de posição do elemento nele.

# PROGRAMMER



# Arrays

Cada elemento do vetor é inicializado a um valor default, dependendo do tipo de dados:

- ▷ *null* - para objetos
- ▷ 0 - para int, long, short, byte, float, double
- ▷ Unicode 0 - para char
- ▷ False - para boolean

O número da posição do vetor começa em 0 e vai até n-1.

Exemplo:

```
int[] c = new int[12];
```

|   |         |      |
|---|---------|------|
| Nome do array (c) →                           | c[ 0 ]  | -45  |
|   | c[ 1 ]  | 6    |
|   | c[ 2 ]  | 0    |
|   | c[ 3 ]  | 72   |
|   | c[ 4 ]  | 1543 |
|   | c[ 5 ]  | -89  |
|   | c[ 6 ]  | 0    |
|   | c[ 7 ]  | 62   |
|   | c[ 8 ]  | -3   |
|   | c[ 9 ]  | 1    |
|   | c[ 10 ] | 6453 |
| Índice (ou subcrito) do elemento no array c → | c[ 11 ] | 78   |

# Arrays

```
//Sintaxe:
Public nomeDaClasse{
    public static void main(String[] args){
        //Declaração:
        int[] vetor = new int[10];

        //Atribuindo valores aos elementos:
        vetor[0] = 10;
        vetor[1] = 20;
        vetor[2] = 15;

        //Acesso aos elementos do array:
        int elemento_1 = vetor[0];
        int elemento_2 = vetor[1];
        int elemento_3 = vetor[2];
    }
}
```

# Arrays

//Exemplo:

```
public static void main(String[] args){  
    int[] array = new int[10];  
    int total = 0;  
    Scanner teclado = new Scanner(System.in);  
  
    for(int i=0; i<10; i++)  
        array[i] = teclado.nextInt();  
  
    for(int i=0; i<10; i++)  
        total += array[i];  
  
    System.out.printf("Soma total é "+total);  
}
```

## Declaração

Vetores podem ser inicializados no momento em que são criados. Sintaxe:

```
String[] semana = {"Dom", "Seg", "Ter", "Qua", "Qui",  
"Sex", "Sab"};
```

```
String[][] usuarios = { {"João", "Ninguém"}, {"Maria",  
"D.", "Aparecida"}, {"Fulano", "de", "Tal"} };
```

Essa inicialização não pode ser usada em outras situações (depois que o vetor já existe), exceto usando new, da forma:

```
String[] semana;
```

```
semana = new String[] {"Dom", "Seg", "Ter", "Qua",  
"Qui", "Sex", "Sab"};
```

# Arrays de objetos

```
public class Conta {  
    public int a;  
    public int b;  
  
    public Conta(int a, int b) {  
        this.a = a;  
        this.b = b;  
    }  
  
    public int Soma() {  
        return a + b;  
    }  
}
```



# Arrays de objetos

```
//Sintaxe:
Public nomeDaClasse{
    public static void main(String[] args){
        //Declaração:
        Conta[] teste = new Conta[10];

        //Inicializando as posições individuais:
        for(int i = 0; i < 10; i++)
            teste[i] = new Conta(i, i+20);

        //Acesso aos elementos e métodos do array:
        System.out.println(teste[3].Soma());
        System.out.println(teste[4].a);
    }
}
```

2.

## Passando arrays para métodos

## Passando arrays para métodos

Para passar um argumento de array para um método, especifique o nome do array sem nenhum colchete.

Por exemplo:

//Array

```
double[] preços = new double[10]
```

//Método

```
alterarPreços(preços);
```

//Declaração do método

```
Void alterarPreços(double[] a){
```

```
...
```

```
}
```



# Arrays

//Exemplo:

```
public static void main(String[] args){
    int[] array = { 1, 2, 3, 4, 5};
    for(int i : array)
        System.out.print(i+" ");
    System.out.print("\n");
    alterarValores(array);

    for(int i : array)
        System.out.print(i+" ");
    }

public void alterarValores(int[] a){
    for(int i : a)
        i *=2;
}
```

3.

# Arrays multidimensionais

# Arrays multidimensionais

Vetores multidimensionais em Java são vetores de vetores

- ▷ É possível criar toda a hierarquia (vetor de vetor de vetor...), para fazer vetores retangulares ...

```
int [][][] prisma = new int [3][2][2];
```

- ▷ ... ou criar apenas o primeiro nível (antes de usar, porém, é preciso criar os outros níveis)

```
int [][][] prisma2 = new int [3][][];  
prisma2[0] = new int[2][];  
prisma2[1] = new int[3][2];  
prisma2[2] = new int[4][4];  
prisma2[0][0] = new int[5];  
prisma2[0][1] = new int[3];
```

# Arrays

//Exemplo:

```
public static void main(String[] args){
    int[][] array1 = {{1,2,3},{4,5,6}};
    int[][] array2 = {{1,2},{3},{4,5,6}};

    System.out.print("Array 1:\n");
    imprimeArray(array1);
    System.out.print("Array 2:\n");
    imprimeArray(array2);
}

public static void imprimeArray(int[][] a){

    for(int[] b: a){
        for(int i: b)
            System.out.print(i+" ");
        System.out.print("\n");
    }
}
```



# DESAFIO

E aí, vamos praticar?



# Fibonacci em Vetor

Faça um programa que leia um valor e apresente o número de Fibonacci correspondente a este valor lido. Lembre que os 2 primeiros elementos da série de Fibonacci são 0 e 1 e cada próximo termo é a soma dos 2 anteriores a ele. Todos os valores de Fibonacci calculados neste problema devem caber em um inteiro de 64 bits sem sinal.

**Entrada:** A primeira linha da entrada contém um inteiro  $T$ , indicando o número de casos de teste. Cada caso de teste contém um único inteiro  $N$  ( $0 \leq N \leq 60$ ), correspondente ao  $N$ -ésimo termo da série de Fibonacci.

**Saída:** Para cada caso de teste da entrada, imprima a mensagem "Fib( $N$ ) =  $X$ ", onde  $X$  é o  $N$ -ésimo termo da série de Fibonacci.

**Entrada:**

3  
0  
4  
2

**Saída:**

Fib(0) = 0  
Fib(4) = 3  
Fib(2) = 1

# Acima da Diagonal Principal

Leia um caractere maiúsculo, que indica uma operação que deve ser realizada e uma matriz  $M[12][12]$ . Em seguida, calcule e mostre a soma ou a média considerando somente aqueles elementos que estão acima da diagonal principal da matriz, conforme ilustrado abaixo (área verde).

[illegible]

# Acima da Diagonal Principal

**Entrada:** A primeira linha de entrada contém um único caractere Maiúsculo O ('S' ou 'M'), indicando a operação (Soma ou Média) que deverá ser realizada com os elementos da matriz. Seguem os 144 valores de ponto flutuante que compõem a matriz.

**Saída:** Imprima o resultado solicitado (a soma ou média), com 1 casa após o ponto decimal.

**Entrada:**

S

1.0

0.0

-3.5

2.5

4.1

...

**Saída:**

12.6

# Obrigado!

## **Alguma pergunta?**

Você pode nos contatar em:  
[ywassef@hotmail.com](mailto:ywassef@hotmail.com)