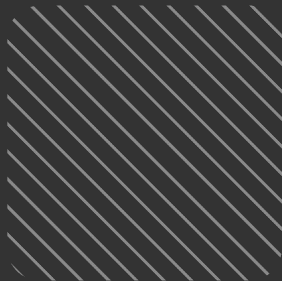


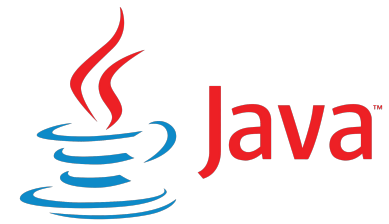
Arrays

// Vetores e Matrizes

IT BOARDING

BOOTCAMP





Índice



01 Arrays (Arranjos)

03 Matrizes

02 Vetores

04 Erros de estouro

IT BOARDING

BOOTCAMP

Arrays

// (Arranjos)

IT BOARDING

BOOTCAMP

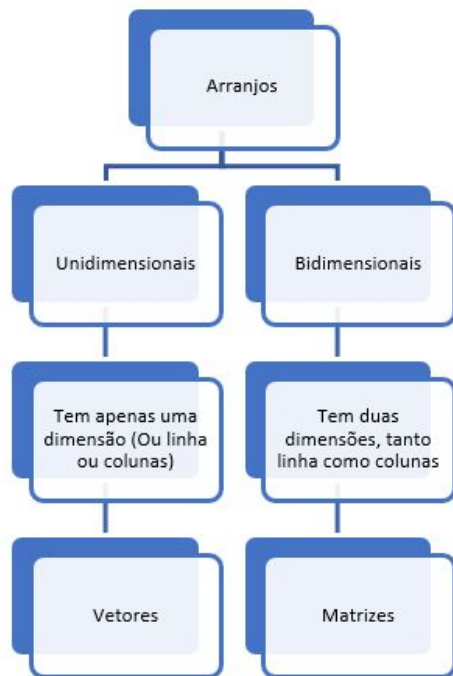




Arranjos

- São um **conjunto de dados** armazenados na memória de forma **contígua** com o **mesmo nome**, mas com **índices** diferentes para identificar cada um deles.
- Eles são **estruturas estáticas**. Uma vez declarados e inicializados, eles devem **manter seu tamanho** durante a execução de um programa.
- Cada array pode ser declarado com apenas **um tipo de dado**, portanto, ele só pode armazenar dados desse tipo.

Tipos de arranjos





Vetores

// Arranjos Unidimensionais

IT BOARDING

BOOTCAMP





Vetores (Arranjos unidimensionais)

- Os vetores são **arranjos unidimensionais** caracterizadas por terem apenas linhas ou colunas.
- Cada posição possui um **índice** que permite sua identificação. A numeração do índice começa em 0 e termina no tamanho da matriz - 1. (Por exemplo: 6 - 1).

| ÍNDICES | | | | | |
|---------|----|----|----|----|-----|
| 0 | 1 | 2 | 3 | 4 | 5 |
| 2 | 32 | 45 | 68 | 21 | 123 |



Vetores (Arranjos unidimensionais)

// Declaração e inicialização

```
{ } int numeros[] = new int [6]; // declaro e inicializo um vetor de 6  
posições
```

// Carga (manual)

```
{ } numeros[0] = 2;  
numeros[1] = 32;  
numeros[2] = 45;  
numeros[3] = 68;  
numeros[4] = 21;  
numeros[5] = 123;
```




Vetores (Arranjos unidimensionais)

// Percorrer

```
//Percorremos um vetor e mostramos o que há em cada posição
for (int i=0; i<numeros.length; i++) {
    System.out.println("Estou no índice: " + i);
    System.out.println("Eu salvei um" + numeros [i]);
}
```



Vetores (Arrays unidimensionales)

// Carga por teclado

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

//percorremos, carregamos cada posição pelo teclado
{} for (int i=0; i<numeros.length; i++) {
    System.out.println("Insira o valor para a posição: " + i);
    numeros[i] = tecla.nextInt();
}
```

Matrizes

IT BOARDING

BOOTCAMP



Matrizes



- Matrizes são arranjos bidimensionais caracterizadas por terem dois índices. Um para lidar com as linhas e outro para lidar com as colunas.

ÍNDICES DAS COLUNAS

012

0

2

32

45

1

68

21

123

2

5

77

69

ÍNDICES DAS LINHAS



Matrizes

// Declaração e inicialização

ÍNDICE DAS LINHAS

```
{}
```

```
int matriz[][] = new int [3][3]; // declaro e inicializo una matriz de  
3 filas y 3 columnas.
```

ÍNDICE DAS COLUNAS

// Carga (manual)

```
{}
```

```
numeros[0][0] = 52;  
numeros[0][1] = 133;  
numeros[0][2] = 24;  
...  
numeros[2][2] = 86;
```



Matrizes

// Percorrer

```
{  
    //nós examinaremos e mostraremos o que está em cada posição  
    //para as matrizes usamos dois índices, um para linhas e um para colunas  
    for (int f=0; f<=2; f++) {  
        for (int c=0; c<=2; c++) {  
            System.out.println("Fila: " + f + "Columna: " + c);  
            System.out.println("Tengo guardado un " + matriz[f][c]);  
        }  
    }  
}
```



Matrizes

// Carga por teclado

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

//nós percorremos e carregamos a matriz ao mesmo tempo
for (int f=0; f<=2; f++) {
    for (int c=0; c<=2; c++) {
        System.out.println("Insira o valor para a posição: " + i);
        matriz[f][c] = tecla.nextInt();
    }
}
```

Erros de estouro

IT BOARDING

BOOTCAMP





Erro de estouro

- Erros de estouro ocorrem quando tentamos nos referir a uma posição em nosso array (seja vetor ou matriz) que não existe ou que está fora dos limites do tamanho que declaramos. Por exemplo:

```
int idades[] = new int [2];  
idades[0] = 15;  
idades[1] = 26;  
idades[2] = 37;
```

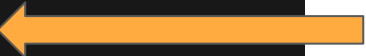
Estou me referindo a uma posição em minha matriz que não existe, pois tem apenas 2 posições (0 e 1)



Erro de estouro

- Erros de estouro ocorrem quando tentamos nos referir a uma posição em nosso array (seja vetor ou matriz) que não existe ou que está dentro de dois limites do tamanho que declaramos. Por exemplo:

```
int idades[] = new int [2];  
idades[0] = 15;  
idades[1] = 26;  
idades[2] = 37;
```



Estou me referindo a uma posição em minha matriz que não existe, mas apenas 2 posições (0 e 1).

Levantando esta exceção:



```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 2 out of bounds for length 2  
at Idade.main(Idade.java:8)
```



Obrigado

IT BOARDING

BOOTCAMP

