

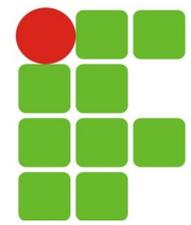
Arrays e Marizes

João Paulo Q. dos Santos
joao.queiroz@ifrn.edu.br



Java

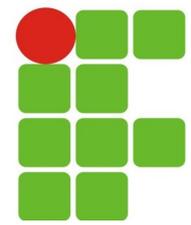




Roteiro

- Arrays e Matrizes em Java:
 - Declaração;
 - Criação;
 - Acesso;

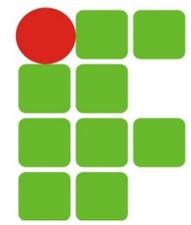




Array: tipos primitivos

- Os arrays (matriz unidimensional) em Java são tratados como objetos e não como uma extensão aos tipos primitivos.
- Propriedades:
 - **Elementos do mesmo tipo;**
 - **Enumerados por índice;**

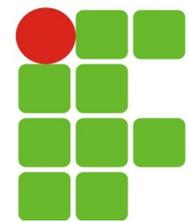
Java



Array (passos)

- Para criar um array em Java deve-se:
 - declarar uma variável que conterà o array;
 - criar um novo objeto array;
 - referenciar a variável ao objeto;
 - armazenar dados no array;

Java



Array (declaração)

- Declaração de um array:

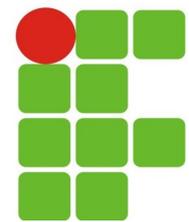
```
<tipo_array>[ ] <nome_variável>;
```

- Onde:

- tipo_array: classe ou tipo primitivo da linguagem;
- nome_variável: identificador válido;

```
int[ ] vetor;
```

Java



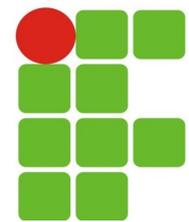
Array (Criação)

- A criação de arrays pode ser feita de duas formas:
 - utilizando new (usado para criação de objetos)

```
int[ ] vetor1 = new int[10];
```
 - Inicializando o array

```
int[ ] vetor1 = {12,3,45,100};
```

Java



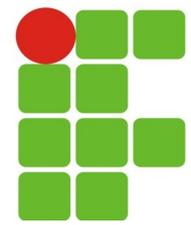
Array (acesso)

- Os índices dos arrays em Java iniciam em 0 até [tamanho - 1], e são verificados em tempo de compilação para que não ultrapassem os limites, pois se isto ocorrer, Java acusará a existência de erro no programa;

```
int[ ] vetor1 = new int[5];
```



Vetor de inteiros com 5 posições
Índice variando de 0 até 4



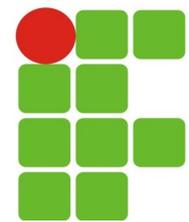
Array acesso

- Acesso aos elementos: <nome_array> [índice]
- Tamanho de um array: <nome_array>.length;

```
int tamanho;  
int[] vetor = new int[5];  
vetor[3] = 20;  
tamanho = vetor.length;
```



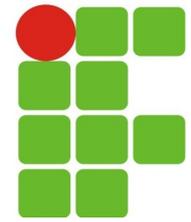
tamanho recebe o valor 5 (que é o tamanho de vetor)



Array Exemplo

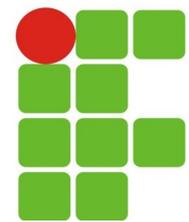
```
package br.edu.ifrn.exemplos;  
public class Array {  
    public static void main (String args[]){  
        double[] numeros = new double[10];  
        for (int i = 0; i < numeros.length; i++) {  
            numeros[i] = Math.random();  
        }  
        for (int i = 0; i < numeros.length; i++) {  
            System.out.println(numeros[i]);  
        }  
    }  
}
```

Declarando um array
Iniciando um array
Acessando um array



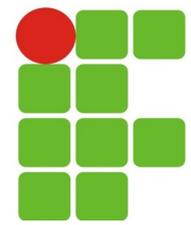
Array Exemplo

```
package br.edu.ifrn.exemplos;
public class Array {
    public static void main (String args[]){
        int [] numeros = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
        for (int i = 0; i < numeros.length; i++) {
            System.out.print(numeros[i]+" ");
        }
    }
}
```



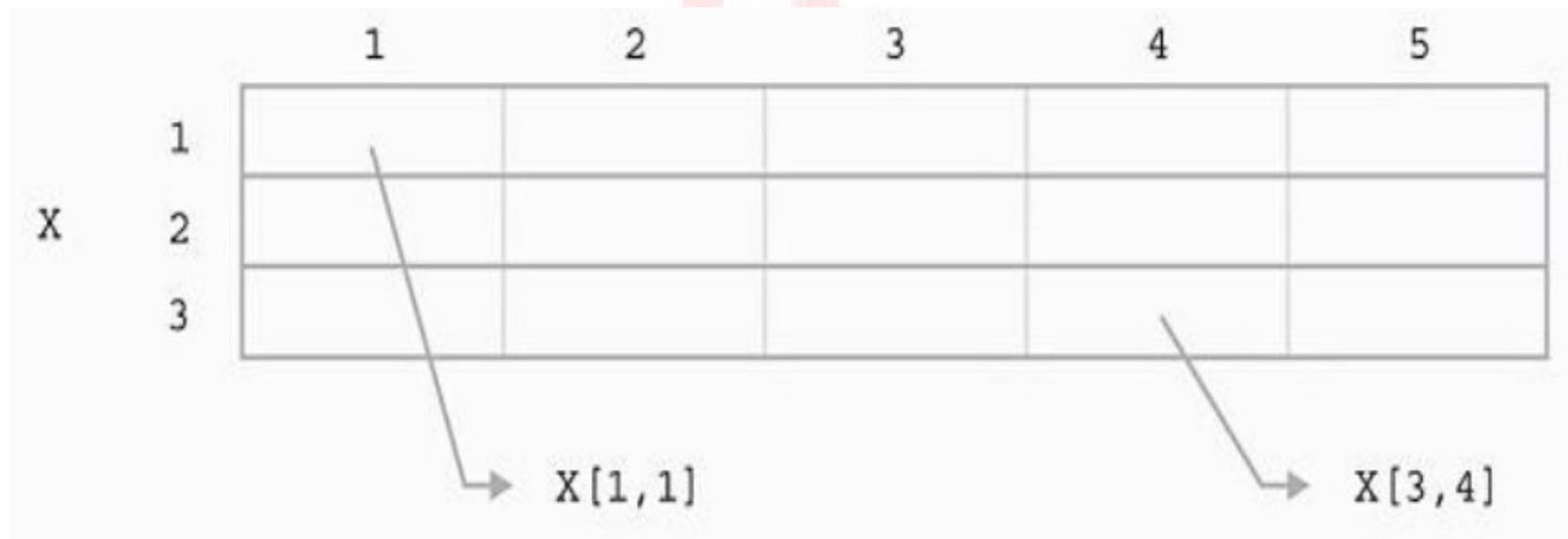
Matriz

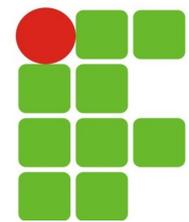
- Uma matriz é uma variável composta homogênea multidimensional;
- Formada por uma sequência de variáveis, todas do mesmo tipo, com o mesmo identificador (mesmo nome), e alocadas sequencialmente na memória;
- Uma vez que as variáveis têm o mesmo nome, o que as distingue são índices que referenciam sua localização dentro da estrutura;
- Uma variável do tipo matriz precisa de um índice para cada uma de suas dimensões.



Matriz

- Define uma matriz bidimensional, onde o tamanho da 1ª dimensão (linha) é 3 e o da 2ª dimensão (coluna) é 5:





Matriz

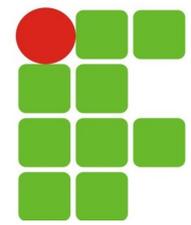
- Declaração da matriz:

```
tipo_dos_dados nome_variável [ ] [ ];
```

```
nome_variável = new tipo_dos_dados [dimensão1] [dimensão2]
```

Onde:

- tipo_dos_dados: é o tipo de dados que poderá ser armazenado na sequência de variáveis que formam a matriz;
- nome_variável: é o nome dado a variável do tipo matriz;
- [dimensão1]: representa o tamanho da primeira dimensão da matriz;
- [dimensão2]: representa o tamanho da segunda dimensão da matriz.



Matriz

- Exemplo:

```
int mat [ ] [ ];
```

```
mat = new int [2] [6];
```

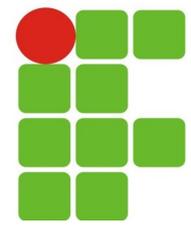
ou

```
int mat [ ] [ ] = new int [2] [6];
```

ou

```
int [ ] [ ] mat = new int [2] [6];
```





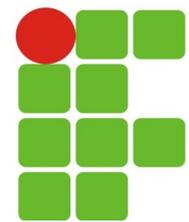
Matriz

- Exemplo:

`mat [1] [5] = 5` → Atribui o valor 5 à posição identificada pelos índices 1 (2ª linha) e 4 (5ª coluna);

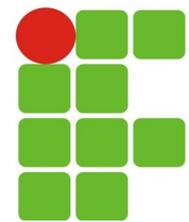
	0	1	2	3	4	5
0						
1					5	

Java



Matriz

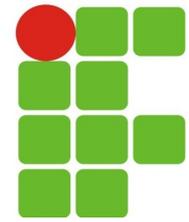
- Preenchendo a matriz:
 - Significa percorrer todos os seus elementos, atribuindo-lhes um valor;
 - Este valor pode ser recebido do usuário, por meio de teclado, ou gerado pelo programa;
- Mostrando a matriz:
 - Pode-se percorrer todos os elementos da matriz, acessando o seu conteúdo;
 - Observe que vão ser utilizados dois índices i e j ;
 - Estes índices estão atrelados a estruturas de repetição que mantêm a variação de ambos dentro de intervalos permitidos;
 - O índice i representa as linhas, varia entre 0 e 1 e o índice j , representa as colunas, varia entre 0 e 5.



Matriz Exemplo

```
package br.edu.ifrn.exmplos;
import java.util.Random;
public class Matriz {
    public static void main(String[] args) {
        int mat [ ][ ] = new int [2][6];
        Random aleatorio = new Random();
        int i, j;
        for (i = 0; i < 2; i++) {
            for (j = 0; j < 6; j++) {
                mat[i][j] = aleatorio.nextInt(10);
            }
        }
        for (i = 0; i < 2; i++) {
            for (j = 0; j < 6; j++) {
                System.out.print(mat[i][j]+" ");
            }
            System.out.println(" ");
        }
    }
}
```

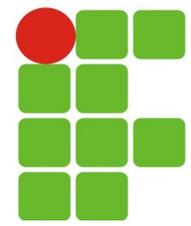
Mesma sintaxe, muda apenas que o vetor têm duas dimensões



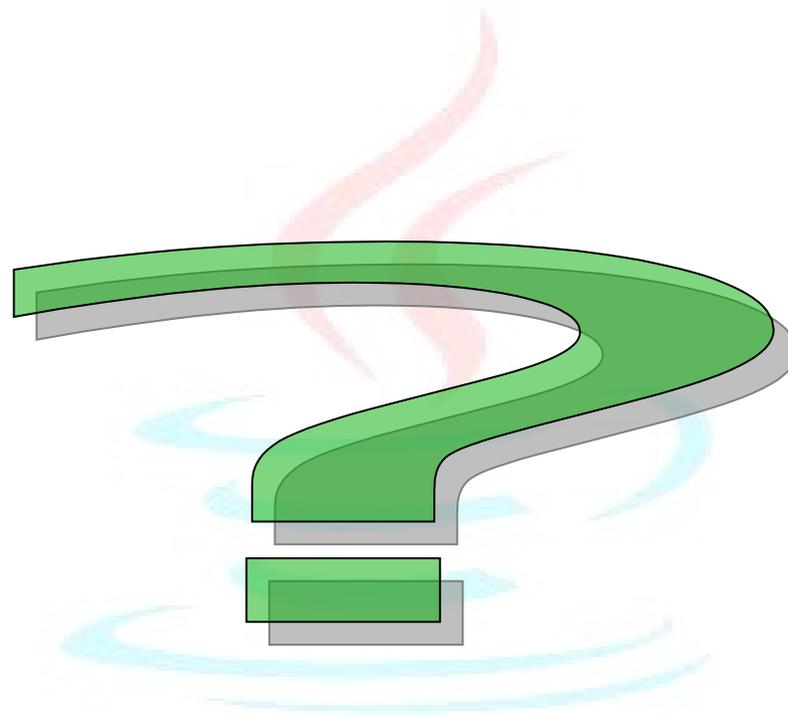
Matriz Exemplo

```
package br.edu.ifrn.exmplos;
import java.util.Scanner;
public class Matriz {
    public static void main(String[] args) {
        int mat [ ] [ ] = new int [2][6];
        Scanner sc = new Scanner(System.in);
        int i, j;
        for (i = 0; i < 2; i++) {
            for (j = 0; j < 6; j++) {
                mat[i][j] = sc.nextInt();
            }
        }
        for (i = 0; i < 2; i++) {
            for (j = 0; j < 6; j++) {
                System.out.print(mat[i][j]+" ");
            }
            System.out.println(" ");
        }
    }
}
```

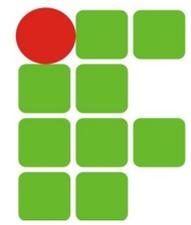
Mesma sintaxe, muda apenas que o vetor têm duas dimensões



Dúvidas



Java



Exercício

Crie um array com dez posições e preencha com valores aleatórios e ordene do menor para o maior

Java