

Arranjos 2D

TIPO DE DADOS
ARRANJO EM JAVA

Concluído
anteriormente...

Foi apresentado a criação e manipulação de Arranjos 1D.

Discutiu-se como os arranjos 1D são armazenados e acessados na memória

Verificou-se que os Arranjos são tratados como objetos em JAVA.

Foram apresentadas duas formas de realizar uma cópia dos Arranjos 1D

Objetivo deste tutorial

Apresentar como são armazenados e manipulados os Arranjos 2D em JAVA por meio de exemplos.

Propor exercícios práticos.

TD Arranjos 2D em JAVA

Para implementar o mapeamento índice x posição, o Java adota o seguinte sistema de índices:

[linha][coluna]

para indicar a posição do primeiro elemento armazenado em um arranjo bidimensional.

		Columns →				
		0	1	2	3	4
Rows ↓	0	5	12	17	9	3
	1	13	4	8	14	1
	2	9	6	3	7	21

2D Array of size 3 x 5

TD Arranjos 2D em JAVA - criação

		Columns →				
		0	1	2	3	4
Rows ↓	0	5	12	17	9	3
	1	13	4	8	14	1
	2	9	6	3	7	21

2D Array of size 3 x 5

```
1 public class Prog010{
2     public static void main(String args[]){
3         float[][] a = {{ 5,12,17, 9, 3},
4                         {13, 4, 8,14, 1},
5                         { 9, 6, 3, 7,21}};
6
7         System.out.println(a[0][1]);
8     }
9 }
```

TD Arranjos 2D em JAVA - criação

		Columns →				
		0	1	2	3	4
Rows ↓	0	5	12	17	9	3
	1	13	4	8	14	1
	2	9	6	3	7	21

2D Array of size 3 x 5

```
1 public class Prog010{
2     public static void main(String args[]){
3         float[][] a = {{ 5,12,17, 9, 3},
4                         {13, 4, 8,14, 1},
5                         { 9, 6, 3, 7,21}};
6
7         System.out.println(a[0][1]);
8     }
9 }
```

```
C:\Users\eduar\Desktop\ED>java Prog010
12.0
```

TD Arranjos 2D em JAVA - criação

```
int[][] a = new int[3][4];
```

	Column 1	Column 2	Column 3	Column 4
Row 1	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Row 2	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Row 3	a[2][0]	a[2][1]	a[2][2]	a[2][3]

TD Arranjos 2D em JAVA - criação

```
int[][] a = {  
    {1, 2, 3},  
    {4, 5, 6, 9},  
    {7},  
};
```

	Column 1	Column 2	Column 3	Column 4
Row 1	1 a[0][0]	2 a[0][1]	3 a[0][2]	
Row 2	4 a[1][0]	5 a[1][1]	6 a[1][2]	9 a[1][3]
Row 3	7 a[2][0]			

TD Arranjos 2D em Java - criação

```
1  class Prog012{
2      public static void main(String[] args) {
3          int[][] a = {
4              {1, 2, 3},
5              {4, 5, 6, 9},
6              {7},
7          };
8          System.out.println("Length of row 1: " + a[0].length);
9          System.out.println("Length of row 2: " + a[1].length);
10         System.out.println("Length of row 3: " + a[2].length);
11     }
12 }
```

TD Arranjos 2D em Java - criação

```
1  class Prog012{
2      public static void main(String[] args) {
3          int[][] a = {
4              {1, 2, 3},
5              {4, 5, 6, 9},
6              {7},
7          };
8          System.out.println("Length of row 1: " + a[0].length);
9          System.out.println("Length of row 2: " + a[1].length);
10         System.out.println("Length of row 3: " + a[2].length);
11     }
12 }
```

```
C:\Users\eduar\Desktop\ED>java Prog012
Length of row 1: 3
Length of row 2: 4
Length of row 3: 1
```

TD Arranjos 2D em JAVA - impressão

```
1  class Prog013{
2      public static void main(String[] args) {
3          int[][] a = {
4              {1, -2, 3},
5              {-4, -5, 6, 9},
6              {7},
7          };
8          for (int i = 0; i < a.length; ++i){
9              for(int j = 0; j < a[i].length; ++j)
10                 System.out.print(a[i][j] + " ");
11                 System.out.println();
12             }
13         }
14     }
```

TD Arranjos 2D em JAVA - impressão

```
1  class Prog013{
2      public static void main(String[] args) {
3          int[][] a = {
4              {1, -2, 3},
5              {-4, -5, 6, 9},
6              {7},
7          };
8          for (int i = 0; i < a.length; ++i){
9              for(int j = 0; j < a[i].length; ++j)
10                 System.out.print(a[i][j] + " ");
11                 System.out.println();
12             }
13         }
14     }
```

```
C:\Users\eduar\Desktop\ED>java Prog013
1 -2 3
-4 -5 6 9
7
```

TD Arranjos 2D em JAVA – Cópias com clone()

```
1 public class Prog011{
2     public static void main(String args[]){
3         float[][] a = {{ 5,12,17, 9, 3},
4                         {13, 4, 8,14, 1},
5                         { 9, 6, 3, 7,21}};
6         float[][] b = a.clone();
7         System.out.println(b[0][1]);
8     }
9 }
```

```
C:\Users\eduar\Desktop\ED>java Prog011
12.0
```

TD Arranjos 2D em JAVA – Cópias com clone()

```
1  public class Prog015{
2      public static void main(String args[]){
3          float[][] a = {{ 5,12,17, 9, 3},
4                          {13, 4, 8,14, 1},
5                          { 9, 6, 3, 7,21}};
6          float[][] b = a.clone();
7          System.out.println(a + " " + b);
8          System.out.println(b[0][1]);
9      }
10 }
```

TD Arranjos 2D em JAVA – Cópias com clone()

```
1  public class Prog015{
2      public static void main(String args[]){
3          float[][] a = {{ 5,12,17, 9, 3},
4                          {13, 4, 8,14, 1},
5                          { 9, 6, 3, 7,21}};
6          float[][] b = a.clone();
7          System.out.println(a + " " + b);
8          System.out.println(b[0][1]);
9      }
10 }
```

```
C:\Users\eduar\Desktop\ED>java Prog015
[[F@15db9742 [[F@6d06d69c
12.0
```

Código base para os exercícios propostos

```
1  public class Exercicios_Arranjo2D{  
2      public static void main(String args[]){  
3          int n = 5; // valor de n pode estar entre 2 e 10  
4          int[][] a = new int[n][n];  
5          // aqui entra o seu código  
6      }  
7  }
```


Exercício proposto 01

Construa um programa em Java que preencha uma matriz quadrada com a soma dos índices linha e coluna de cada posição.

Ao final o programa deve apresentar a matriz.

Exercício proposto 02

Construa um programa em Java que preencha uma matriz identidade

$$I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ é a matriz identidade de ordem 3.}$$

$$I_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ é a matriz identidade de ordem 4.}$$

$$I_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \text{ é a matriz identidade de ordem 5.}$$

Ao final o programa deve apresentar a matriz.

Exercício proposto 03

4. As seguintes matrizes têm o apelido de **matrizes-espírais** de dimensões 5 e 6:

21	22	23	24	25
20	7	8	9	10
19	6	1	2	11
18	5	4	3	12
17	16	15	14	13

e

21	22	23	24	25	26
20	7	8	9	10	27
19	6	1	2	11	28
18	5	4	3	12	29
17	16	15	14	13	30
36	35	34	33	32	31

Fazer um programa para ler **n** e construir uma matriz-espíral de dimensão **n**

Conclusões

Foi apresentado os Arranjos 2D.

Apresentou-se algumas formas de criação de um Arranjo 2D.

Verificou-se que a cópia por meio do método clone() funciona.

Foram propostos exercícios.