

LINGUAGENS DE PROGRAMAÇÃO (JAVA)

Aula 2 — Arrays de Duas Dimensões (Matrizes)

Organizando dados em linhas e colunas

Unidade: Arrays em Java | 2 horas



Objetivos da aula



Criar e acessar

Criar matrizes e acessar/alterar elementos com `matriz[linha][coluna]`.



Percorrer com for duplo

Usar dois laços aninhados para visitar todos os elementos.



Aplicar em problemas reais

Somar, contar, comparar e organizar dados como notas de alunos.

Vetor × matriz

O vetor guarda valores em uma única linha, com um índice. A matriz organiza valores em linhas e colunas, com dois índices.

Critério	Vetor	Matriz
Dimensões	1	2
Índices	1 índice	2 índices: linha e coluna
Declaração	<code>int[] v = new int[5];</code>	<code>int[][] m = new int[3][3];</code>
Acesso	<code>v[i]</code>	<code>m[linha][coluna]</code>

Criando uma matriz em Java

```
// Declarar e criar em duas etapas
int[][] matriz;
matriz = new int[3][3];

// Ou tudo em uma única linha
int[][] matriz = new int[3][3];

// matriz.length      -> 3 (linhas)
// matriz[i].length    -> 3 (colunas)
```

Linha, coluna e acesso a um elemento

	Coluna 0	Coluna 1	Coluna 2
Linha 0	10	20	30
Linha 1	40	50	60
Linha 2	70	80	90

`matriz[0][0]`

linha 0, coluna 0 → 10

índices sempre começam em zero

REGRA PRÁTICA

**O primeiro for percorre as linhas.
O segundo for percorre as colunas.**

É por isso que, para percorrer uma matriz inteira, sempre usamos dois laços for aninhados.

Percorrendo e exibindo a matriz

```
for (int i = 0; i < matriz.length; i++) {  
    for (int j = 0; j < matriz[i].length; j++) {  
        System.out.print(matriz[i][j] + " ");  
    }  
    System.out.println();  
}  
  
// i -> controla a linha    |    j -> controla a coluna
```

Preenchendo uma matriz com Scanner

```
Scanner entrada = new Scanner(System.in);
int[][] matriz = new int[3][3];

for (int i = 0; i < matriz.length; i++) {
    for (int j = 0; j < matriz[i].length; j++) {
        System.out.print("Linha " + i + ", coluna " + j + ": ");
        matriz[i][j] = entrada.nextInt();
    }
}
```

Operações comuns sobre matrizes



Soma total

Acumular todos os elementos em uma variável soma.



Pares e ímpares

Combinar o percurso com if (`matriz[i][j] % 2 == 0`).



Maior e menor

Comparar cada elemento com o maior/menor já encontrado.

Soma de cada linha e de cada coluna

```
// Soma de cada LINHA: zera a soma no for externo
for (int i = 0; i < matriz.length; i++) {
    int soma = 0;
    for (int j = 0; j < matriz[i].length; j++) soma += matriz[i][j];
    System.out.println("Linha " + i + ": " + soma);
}

// Soma de cada COLUNA: inverte a ordem dos índices
for (int j = 0; j < matriz[0].length; j++) {
    int soma = 0;
    for (int i = 0; i < matriz.length; i++) soma += matriz[i][j];
    System.out.println("Coluna " + j + ": " + soma);
}
```

Exemplo — notas dos alunos

Cada linha representa um aluno; cada coluna representa uma prova. A média de cada aluno é a soma da linha dividida pelo número de provas.

	P1	P2	P3	Média	Situação
Aluno 0	8	7	9	8.0	Aprovado
Aluno 1	6	8	7	7.0	Aprovado
Aluno 2	9	9	10	9.3	Aprovado
Aluno 3	5	6	7	6.0	Aprovado

Diagonal principal

Elementos em que linha = coluna: `matriz[i][i]`.

	Coluna 0	Coluna 1	Coluna 2
Linha 0	10	20	30
Linha 1	40	50	60
Linha 2	70	80	90

```
int n = matriz.length;
for (int i = 0; i < n; i++) {
    System.out.println(
        matriz[i][i]);
}
// Soma: 10+50+90=150
```

Matriz quadrada × matriz retangular

Critério	Quadrada	Retangular
Linhas × colunas	iguais (ex.: 3×3, 4×4)	diferentes (ex.: 2×5, 3×5)
Exemplo	<code>new int[4][4]</code>	<code>new int[3][5]</code>
Diagonais	principal e secundária existem	não se aplicam no sentido tradicional

Resumo para levar da aula

- Matriz = array de duas dimensões: `int[][] m = new int[linhas][colunas]`.
- Acesso e alteração: `matriz[linha][coluna]`. Índices começam em zero.
- O primeiro for percorre linhas; o segundo percorre colunas.
- `matriz.length = linhas`; `matriz[i].length = colunas da linha i`.
- Soma, contagem, maior/menor, soma por linha/coluna e diagonais usam o mesmo esqueleto de for duplo.





Obrigado!

Próxima aula: exercícios de fixação e correção coletiva