

TEMA 6: Vetores e Matrizes em Java

Material Didático para Aula Prática (Console Application)

1. Objetivos

- Entender o conceito de **vetores (arrays unidimensionais)**
 - Compreender **matrizes (arrays bidimensionais)**
 - Aplicar vetores e matrizes em um exemplo prático **sem banco de dados**
 - Visualizar resultados no **console do NetBeans**
-

2. Conceitos Teóricos

2.1 Vetores (Arrays Unidimensionais)

- **O que é?** Uma estrutura que armazena vários valores **do mesmo tipo** em uma única variável.
- **Sintaxe:**
`tipo[] nomeArray = new tipo[tamanho];`
- **Exemplo:**
`int[] idades = new int[5]; // Vetor de 5 posições`

2.2 Matrizes (Arrays Bidimensionais)

- **O que é?** Um "vetor de vetores" que organiza dados em **linhas e colunas** (como uma tabela).
 - **Sintaxe:**
`tipo[][] nomeMatriz = new tipo[linhas][colunas];`
 - **Exemplo:**
`String[][] pessoas = new String[3][4]; // 3 linhas (pessoas) x 4 colunas (dados)`
-

3. Exemplo Prático (NetBeans - Console)

Aplicação: Relatório de Pessoas (sem SQLite)

3.1 Classe Principal (Main.java)

```
public class Main {  
  
    public static void main(String[] args) {  
        // Dados pré-cadastrados em uma MATRIZ (simulando um banco de dados)  
        String[][] dadosPessoas = {  
            {"1", "Joao", "25", "2500.00"}, // ID, Nome, Idade, Salário
```

```

        {"2", "Maria", "30", "3000.00"},
        {"3", "Carlos", "35", "4000.00"}
    };

    // 1. Exibir dados brutos
    System.out.println("=== DADOS BRUTOS ===");
    for (String[] pessoa : dadosPessoas) {
        System.out.println(
            "ID: " + pessoa[0] + ", "
            + "Nome: " + pessoa[1] + ", "
            + "Idade: " + pessoa[2] + ", "
            + "Salario: " + pessoa[3]
        );
    }

    // 2. Calcular estatísticas usando VETORES
    double[] salarios = new double[dadosPessoas.length];
    int[] idades = new int[dadosPessoas.length];

    for (int i = 0; i < dadosPessoas.length; i++) {
        idades[i] = Integer.parseInt(dadosPessoas[i][2]); // Converte String para int
        salarios[i] = Double.parseDouble(dadosPessoas[i][3]); // Converte String para
double
    }

    // 3. Gerar relatório em uma nova MATRIZ
    String[][] relatorio = new String[5][4]; // 5 linhas, 4 colunas

    // Cabeçalho
    relatorio[0][0] = "=== RELATORIO ===";
    relatorio[1][0] = "Media de Idade: " + calcularMedia(idades);
    relatorio[2][0] = "Media Salarial: R$" + calcularMedia(salarios);
    relatorio[3][0] = "Maior Salario: R$" + encontrarMaior(salarios);

    // Linha de totais
    relatorio[4][0] = "TOTAL SALARIOS: R$" + calcularSoma(salarios);

    // 4. Exibir relatório
    System.out.println("\n=== RELATORIO FINAL ===");
    for (String[] linha : relatorio) {
        if (linha[0] != null) {
            System.out.println(linha[0]);
        }
    }
}

// Métodos auxiliares (cálculos)
public static double calcularMedia(double[] valores) {
    double soma = 0;
    for (double valor : valores) {
        soma += valor;
    }
}

```

```

        return soma / valores.length;
    }

    public static double calcularMedia(int[] valores) {
        int soma = 0;
        for (int valor : valores) {
            soma += valor;
        }
        return soma / valores.length;
    }

    public static double encontrarMaior(double[] valores) {
        double maior = valores[0];
        for (double valor : valores) {
            if (valor > maior) {
                maior = valor;
            }
        }
        return maior;
    }

    public static double calcularSoma(double[] valores) {
        double soma = 0;
        for (double valor : valores) {
            soma += valor;
        }
        return soma;
    }
}

```

4. Saída no Console

Ao executar o programa, o resultado será:

```

=== DADOS BRUTOS ===
ID: 1, Nome: João, Idade: 25, Salário: 2500.00
ID: 2, Nome: Maria, Idade: 30, Salário: 3000.00
ID: 3, Nome: Carlos, Idade: 35, Salário: 4000.00

=== RELATÓRIO FINAL ===
=== RELATÓRIO ===
Média de Idade: 30.0
Média Salarial: R$3166.6666666666665
Maior Salário: R$4000.0
TOTAL SALÁRIOS: R$9500.0

```
