



Java Standard Edition (JSE)

Capítulo 03. Arrays



Esp. Márcio Palheta

MSN: marcio.palheta@hotmail.com



Agenda

- Revisão do aula anterior;
- Motivação – minimize riscos;
- Arrays
- Etapas do array
- Formas de inicialização;
- Técnicas para percorrer um array;
- Arrays multidimensionais;
- Exercícios;



Revisão

- Declaração de variáveis;
- Atribuição de valores;
- Casting e comparação de variáveis;
- Controle de fluxos com if e else;
- Laços de repetição for e while;
- Controle de fluxos com break e continue;
- Dúvidas ?



Motivação – Minimize riscos

- O mundo empresarial é extremamente dinâmico e competitivo.
- O técnico bem preparado pode fazer a diferença na hora de tomar decisões críticas ao negócio da empresa;
- Prepare-se a fim de minimizar o risco de cair nas armadilhas do mundo corporativo;
- Vídeos:
 - 03.01 A primeira decepção de EVA;
 - 03.02 Satellite TV System



Novos recursos a aprender

- Declarar e instanciar arrays;
- Popular e percorrer arrays;
- Manipular arrays multidimensionais;



O problema

- Dentro de um bloco, podemos declarar variáveis e usá-las.
 - `int idade1;`
 - `int idade2;`
 - `int idade3;`
 - `int idade4;`
- Mas também podemos declarar um **vetor (array)** de inteiros:
 - `int[] idades;`



Arrays

- Reduzem o número de variáveis;
- Dinamismo para controle de grupos de variáveis;
- Etapas:
 - **Declaração:** Etapa em que a referência do array é declarada;
 - **Construção:** Aqui é definido o tamanho e instanciado o array;
 - **Inicialização:** Os elementos são inseridos no array.



Etapas do Array

- 1. **int[]** jogoSena; **//Declaração**
- 2. jogoSena = **new int**[6]; **//Criação**
- 3. jogoSena[0] = 23; **//Inicialização** da posição 0
- 4. jogoSena[1] = 12; **//Inicialização** da posição 1
- 5. jogoSena[2] = 55; **//Inicialização** da posição 2
- 6. jogoSena[3] = 02; **//Inicialização** da posição 3
- 7. jogoSena[4] = 07; **//Inicialização** da posição 4
- 8. jogoSena[5] = 19; **//Inicialização** da posição 5

- 9. jogoSena[6] = 01; **//OPS...**
 - Java.lang.ArrayIndexOutOfBoundsException



Outra forma de inicialização

- **Sintaxe:**

- `tipoDoArray [] nomeDoArray = {lista de valores}`

- **Exemplo:**

- `int[] outroJogoSena = {23, 12, 55, 02, 07, 19};`
- O que aconteceu?



Percorrer um array

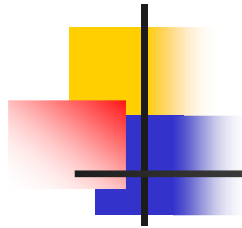
- ```
public static void main(String args[]) {
 int[] idades = new int[10];
 for (int i = 0; i < 10; i++) {
 idades[i] = i * 10;
 }
 for (int i = 0; i < idades.length; i++) {
 System.out.println(idades[i]);
 }
}
```
- Qual a diferença entre as condições de **parada**?



# Facilidade do Java 5.0

---

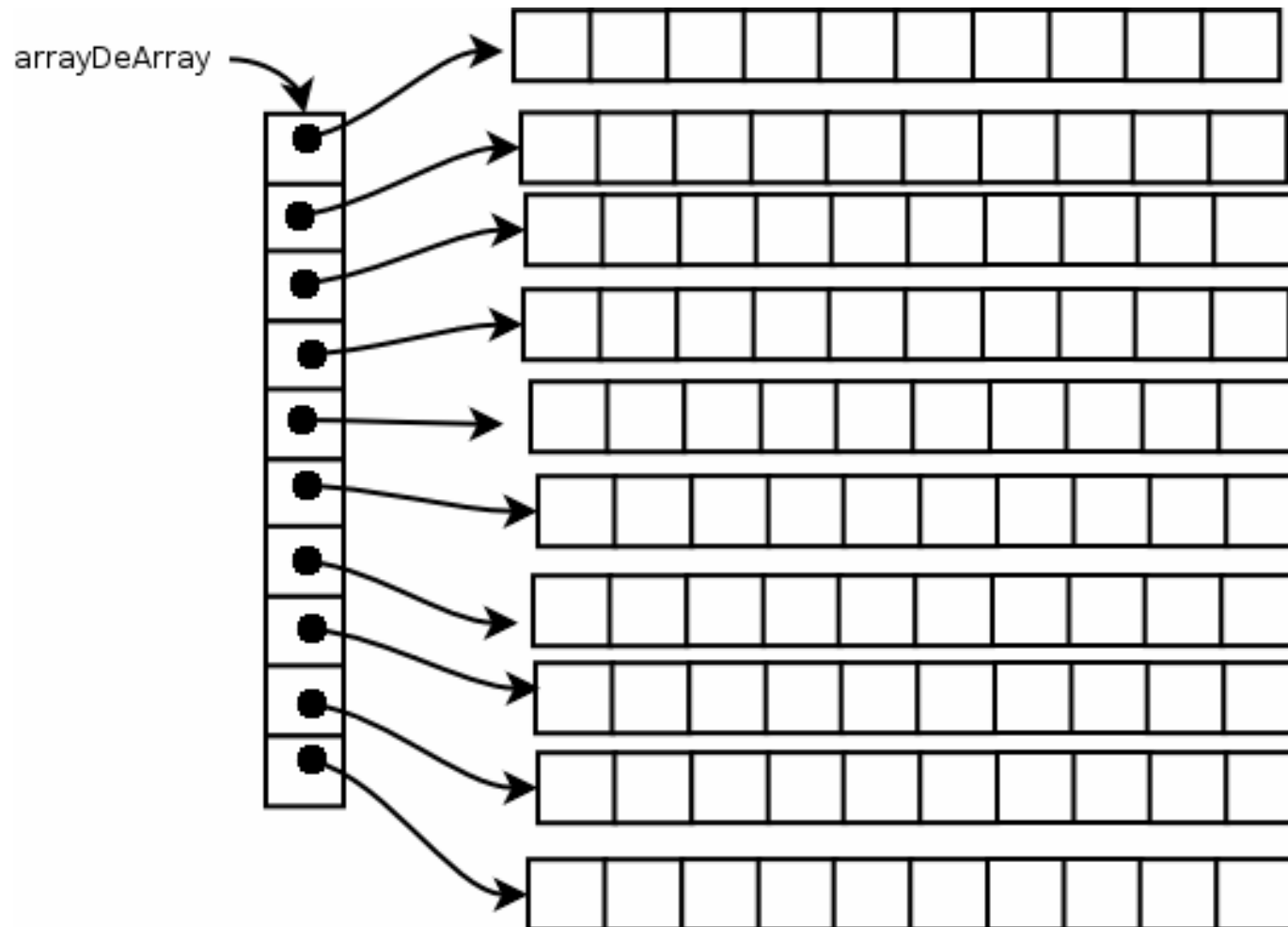
- ```
public static void main(String args[]) {  
    int[] idades = new int[10];  
    for (int i = 0; i < 10; i++) {  
        idades[i] = i * 10;  
    }  
    for (int x : idades) {  
        System.out.println(x);  
    }  
}
```



Arrays multidimensionais

- São estruturas de dados um pouco mais complexas que os arrays estudados até aqui;
- São considerados "Arrays de Arrays"
- O java permite a criação de arrays com qualquer número de dimensões, mas dificilmente há a ocorrência de arrays com mais de três dimensões;

Arrays bidimensionais





Declaração de arrays multidimensionais

- Sintaxe:

- `tipoVariavel [][] nomeVariavel;`

- Exemplo:

- `String jogoDaVelha = new String[3][3];`
- Este exemplo cria um array multidimensional (matriz) com 3 linhas e 3 colunas;



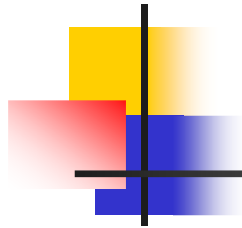
Exercício 01

- Escreva um programa JAVA que:
- solicite do usuário o tamanho de um array de números inteiro;
- solicite o valor de cada elemento do array;
- Calcule a soma e a média dos valores informados;
- Imprima seu tamanho, soma e média de valores, além do conteúdo do array;



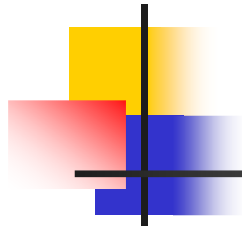
Exercício 02

- Escreva um programa que ordene um array com 10 números inteiros. O algoritmo deve efetuar uma comparação de cada elemento com o seu sucessor e se a ordem não for crescente os elementos devem ter suas posições trocadas, este processo deve se repetir até que o array esteja ordenado. A cada alteração de posições, imprima a ordem atual do array;
 - **int**[] arrayInteiros = {55, 27, 33, 45, 92, 100, 3, 8, 11, 70};



Exercício 03

- Escreva um programa java em que o usuário informe o conteúdo de duas matrizes(A e B) com dimensões 2x2;
- Gere uma matriz C com o resultado da soma entre as matrizes A e B.
- Imprima as matrizes A, B e C;
- OBS: Na soma de matrizes, cada elemento da matriz $A(i,j)$ deve ser somado ao respectivo $B(i,j)$;



Exercício 04

- Escreva um programa que construa uma matriz de double 4x4 e, para cada célula, atribua o valor $i*j$. Imprima o valor de cada elemento da matriz;



Exercícios da lista

- Tempo para implementação de exercícios da lista;
- Tire suas dúvidas;
- Ajude outros colegas com a implementação;
- Aproveite o tempo para praticar;
- Resolva o problema por partes;



Bibliografia

- Java - Como programar, de Harvey M. Deitel
- Use a cabeça! - Java, de Bert Bates e Kathy Sierra
- (Avançado) Effective Java Programming Language Guide, de Josh Bloch



Referências WEB

- SUN: www.java.sun.com

Fóruns e listas:

- Javaranch: www.javaranch.com
- GUJ: www.guj.com.br

Apostilas:

- Argonavis: www.argonavis.com.br
- Caelum: www.caelum.com.br



Java Standard Edition (JSE)

Capítulo 03. Arrays



Esp. Márcio Palheta

MSN: marcio.palheta@hotmail.com