



## Semântica de Referência e Vetores / Matrizes

Prof. Edwar Saliba Júnior  
Fevereiro de 2011



## Conteúdo

- Semântica de Java
- Referência entre Objetos
- Perda de Referência
- Vetores
  - Vetores de Tipos Primitivos
  - Vetores de Objetos
- Vetores Multidimensionais



## Semântica de Referência

- Diferente de outras linguagens de programação (tipo linguagem C, Pascal e etc.), em Java não existem ponteiros;
- Java implementa o conceito de semântica de referência;
- A declaração de um objeto de uma classe Veículo, cria uma referência para um objeto desta classe;
- Exemplo de código em Java: `Veiculo carro;`
- Um objeto criado, deve ser associado a uma referência, ou seja, uma posição de memória:
- Exemplo: `carro = new Veiculo();`

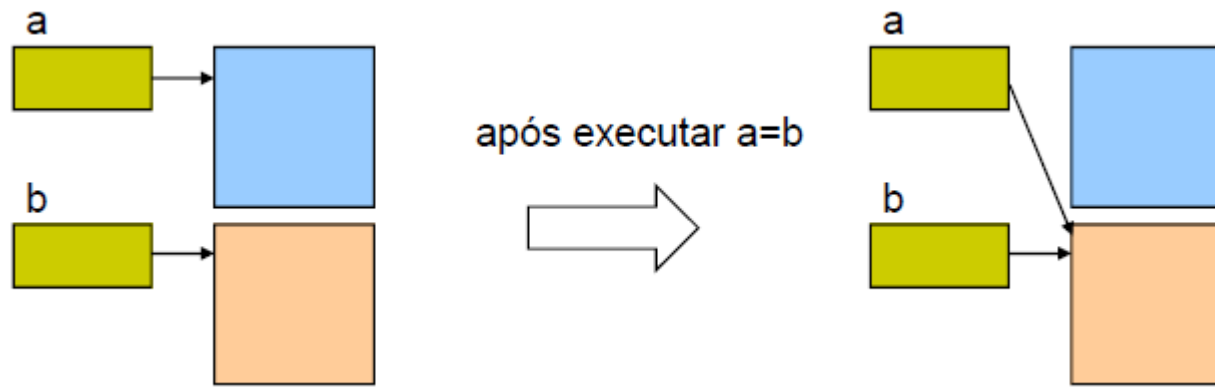
## Referência entre Objetos

- A atribuição de uma referência de um objeto *b* a um objeto *a*, resulta em *a* e *b* referenciando o mesmo objeto:

```
a = new Veiculo();
```

```
b = new Veiculo();
```

```
a = b;
```





## Perda de Referência

- Ao perder a referência, um objeto está fadado a destruição, ou seja, sua eliminação da memória do computador;
- A linguagem Java possui o *Garbage Collector* (Coletor de Lixo), que automaticamente elimina todos os objetos que perderam suas referências, e que ainda estão na memória.

## Exercício

- Determine o que será impresso pelo programa a seguir:

```
1 package classeDoisValores;
2
3 class ClasseDoisValores {
4
5     private int dado1;
6     private int dado2;
7
8     public ClasseDoisValores(int d1, int d2) {
9         dado1 = d1;
10        dado2 = d2;
11    }
12
13    public int getDado1() {
14        return(dado1);
15    }
16
17    public int getDado2() {
18        return(dado2);
19    }
20 }
21
22 public class Main {
23
24     public static void main(String[] args) {
25         ClasseDoisValores objA, objB;
26
27         objA = new ClasseDoisValores(10, 20);
28         objB = new ClasseDoisValores(50, 100);
29
30         System.out.println(objA.getDado1() + " " + objA.getDado2());
31         System.out.println(objB.getDado1() + " " + objB.getDado2());
32
33         objA = new ClasseDoisValores(30, 60);
34
35         System.out.println(objA.getDado1() + " " + objA.getDado2());
36
37         objB = objA;
38
39         System.out.println(objB.getDado1() + " " + objB.getDado2());
40     }
41 }
```

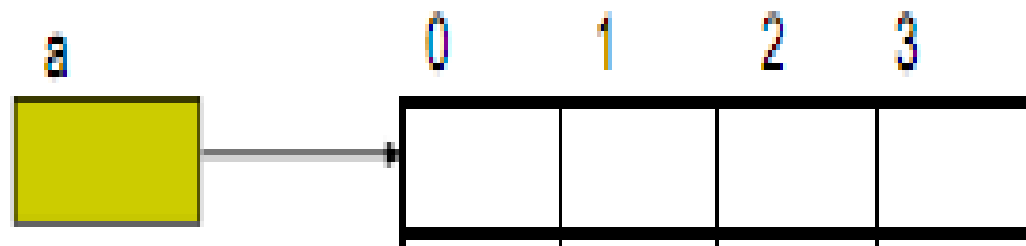
## Vetores

- A declaração de vetores em Java, se dá através do uso de [ ] (colchetes). Onde:
  - Quando os [ ] são utilizados junto ao tipo, significa que todas as variáveis declaradas são vetores. Exemplo: `int[] a, b;`
  - Quando os [ ] são utilizados junto a variável, significa que só a variável é um vetor. Exemplo: `int a[], b;`



## Vetores de Tipos Primitivos

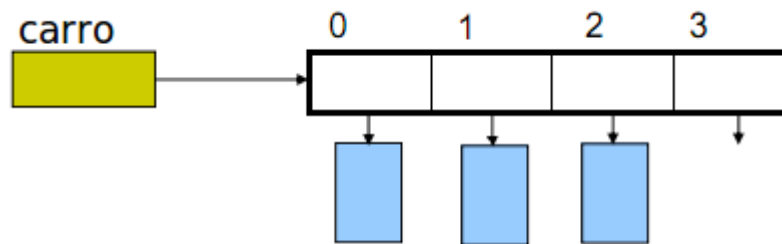
- Um vetor sempre começa pela posição zero;
- Vetores têm que ser instanciados:
  - `a = new int[4];`





## Vetores de Objetos

- Para a criação de vetores de objetos, faz-se necessário a instanciação do vetor em si, e também, a instanciação de cada membro que preencherá as posições do vetor;
- Exemplo:
  - `Veiculo[] carros;`
  - `carros = new Veiculo[4];`
  - `carros[0] = new Veiculo("Fiat", "Uno");`
  - `carros[1] = new Veiculo("Fiat", "Palio");`
  - ...



## Exemplo

```
16 public static void main(String[] args) {
17     int[] a = new int[10], b;
18     int c[], d;
19     b = new int[10];
20     c = new int[5];
21     d = new int[3]; /* Este comando dá erro de compilação
22                        porque d não é um vetor. */
23     a[0] = 10;
24     a[9] = 20;
25     a[10] = 5; /* Este comando gera a exceção
26                  ArrayIndexOutOfBoundsException */
27
28     for (int i = 0; i < a.length; i++) {
29         System.out.print(a[i] + " - ");
30     }
31 }
```



## Vetores Multidimensionais

- Pode-se criar, por exemplo, vetores bidimensionais, em que o número de colunas sejam diferentes para cada linha;
- Exemplo:
  - `int vet[][];`
  - `vet = new int[2][]; // Duas linhas são criadas.`
  - `vet[0] = new int[3]; // Linha 0 terá três colunas.`
  - `vet[1] = new int[7]; // Linha 1 terá sete colunas.`



## Vetores Multidimensionais

- Declaração:
  - `int vetor[];`
  - `int matriz[][];`
  - `int cubo[][][];`
- Criação:
  - `vetor = new int[4];`
  - `matriz = new int[5][7];`
  - `cubo = new int[8][3][15];`



## Exercício

- Construa um *software* onde o usuário seja capaz de gerenciar uma concessionária de carros usados. Este *software* deverá ter a capacidade de armazenamento (em memória principal) de 100 veículos, e deverá possibilitar o cadastro, alteração, exclusão e consulta dos veículos cadastrados. O *software* deverá também, ter a opção de emitir, na tela do computador, um relatório com todos os veículos cadastrados.



## Bibliografia

- DEITEL, H. M.; DEITEL, P. J. **Java Como Programar**; tradução Edson Furmankiewicz; revisão técnica Fábio Lucchini. 6a. ed., São Paulo: Pearson, 2005.
- FERREIRA, Kecia Aline Marques. *Slides da disciplina de Programação de Computadores II*. CEFET-MG, 2009.
- JAVA. Wikipedia – a enciclopédia livre. Disponível em: [http://pt.wikipedia.org/wiki/Java\\_%28linguagem\\_de\\_programa%C3%A7%C3%A3o%29](http://pt.wikipedia.org/wiki/Java_%28linguagem_de_programa%C3%A7%C3%A3o%29) Acesso em: 23 jan. 2011.
- MIZRAHI, Victorine Viviane. **Treinamento em Linguagem C++ - Módulo 2**. 2. ed., São Paulo: Pearson, 2006.