

Programação Orientada a Objetos

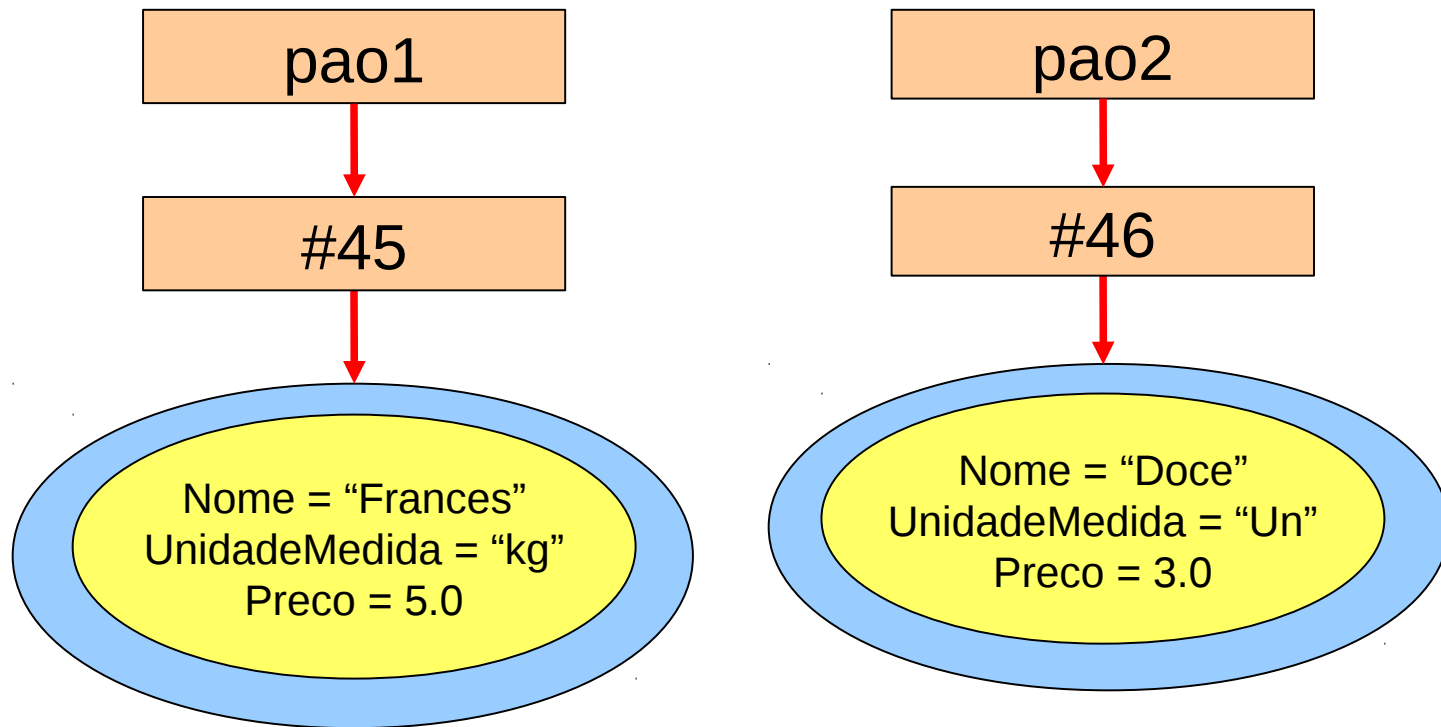
Prof. Evandro César Freiburger

Membros Estáticos – Atributos Estáticos

- Na grande maioria das vezes os atributos são de instâncias (de objetos);
- Isso significa que para cada objeto criado, será gerado sua própria cópia de dados;
- Atributos de estáticos são atributos de classe
 - » Não há a necessidade de mais de uma cópia;
 - » Compartilhamento de dados entre os objetos da mesma classe;
- Veja ilustração a seguir:

Membros Estáticos – Atributos Estáticos

- Cada instância possui sua própria cópia dos dados
- Atributos (nome, unidadeMedida e preco) são de instância;



Membros Estáticos – Atributos de Classe

- Produto6 possui quatro atributos de instância e dois atributos de classe
- Quando a classe for carregada para a memória, serão alocados espaços para os atributos de classe;

```
Produto6.java x
1 public class Produto6 {
2
3     String nome;
4     String unidadeMedida;
5     float preco;
6     int estoque;
7     static final float PRECO_MINIMO=0.1f;
8     static int estoqueGeral;
9
10    public Produto6() {
11        this.unidadeMedida = "Kg";
12        this.preco = PRECO_MINIMO;
13    }
14
15    public Produto6(String nome) {
16        this();
17        this.nome = nome;
18    }
19
20    void adicionarEstoque(int quantidade){
21        estoque += quantidade;
22        estoqueGeral += quantidade;
23    }
24
25
26    float totalizar(final float peso){
27        return this.preco * peso;
28    }
29 }
```

Membros Estáticos – Atributos de Classe

```
public static void main(String[] args) {  
    Produto6 pro1 = new Produto6("Pão de Queijo");  
    Produto6 pro2 = new Produto6("Pão Francês");  
    Produto6 pro3 = new Produto6("Pão Doce");  
    pro3.unidadeMedida = "Un";  
  
    System.out.println("-----");  
    System.out.println("Nome: "+pro1.nome);  
    System.out.println("Estoque: "+pro1.estoque);  
    System.out.println("Estoque Geral: "+pro1.estoqueGeral);  
    pro1.adicionarEstoque(10);  
    System.out.println("-----");  
    System.out.println("Nome: "+pro1.nome);  
    System.out.println("Estoque: "+pro1.estoque);  
    System.out.println("Estoque Geral: "+pro1.estoqueGeral);  
    pro2.adicionarEstoque(5);  
    System.out.println("-----");  
    System.out.println("Nome: "+pro2.nome);  
    System.out.println("Estoque: "+pro2.estoque);  
    System.out.println("Estoque Geral: "+pro2.estoqueGeral);  
    pro3.adicionarEstoque(7);  
    System.out.println("-----");  
    System.out.println("Nome: "+pro3.nome);  
    System.out.println("Estoque: "+pro3.estoque);  
    System.out.println("Estoque Geral: "+pro3.estoqueGeral);  
    System.out.println("-----");  
    System.out.println("Estoque: "+pro1.estoque);  
    System.out.println("Estoque: "+pro2.estoque);  
    System.out.println("Estoque: "+pro3.estoque);  
    System.out.println("Estoque Geral: "+pro1.estoqueGeral);  
    System.out.println("Estoque Geral: "+pro2.estoqueGeral);  
    System.out.println("Estoque Geral: "+pro3.estoqueGeral);  
}
```

```
-----  
Nome: Pão de Queijo  
Estoque: 0  
Estoque Geral: 0  
-----
```

```
Nome: Pão de Queijo  
Estoque: 10  
Estoque Geral: 10  
-----
```

```
Nome: Pão Francês  
Estoque: 5  
Estoque Geral: 15  
-----
```

```
Nome: Pão Doce  
Estoque: 7  
Estoque Geral: 22  
-----
```

```
Estoque: 10  
Estoque: 5  
Estoque: 7  
Estoque Geral: 22  
Estoque Geral: 22  
Estoque Geral: 22
```

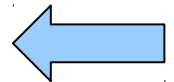
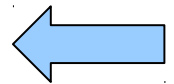
Membros Estáticos – Métodos de Classe

- Da mesma forma que podemos definir atributos de classe é possível definir métodos de classe;
- É útil para:
 - Executar ações que não precisam de instâncias;
 - Manipular atributos de classe sem criar instâncias;
- Existem várias classes do Java que usam esse recurso:
 - `System.out.print`, `System.exit()`
 - `JOptionPane.showMessageDialog()`
 - `public static void main(String args[])`

Membros Estáticos – Métodos de Classe

- Métodos de classes somente podem manipular atributos de classe;
- Métodos de instância podem manipular atributos de instância e atributos de classe;

```
1 public class Produto6 {
2     String nome;
3     String unidadeMedida;
4     float preco;
5     int estoque;
6     static final float PRECO_MINIMO=0.1f;
7     static int estoqueGeral;
8
9     public Produto6(String nome) {
10         this.unidadeMedida = "Kg";
11         this.preco = PRECO_MINIMO;
12         this.nome = nome;
13     }
14     void adicionarEstoque(int quantidade){
15         estoque += quantidade;
16         estoqueGeral += quantidade;
17     }
18
19     static void zerarEstoqueGeral(){
20         estoqueGeral = 0;
21     }
22
23     static int obterEstoqueGeral(){
24         return estoqueGeral;
25     }
26     float totalizar(final float peso){
27         return this.preco * peso;
28     }
29 }
```



Membros Estáticos – Métodos de Classe

```
1 public class Aplicacao7 {
2     public static void main(String[] args) {
3
4         System.out.println("-----");
5         System.out.println("Estoque Geral: "+Produto6.obterEstoqueGeral());
6
7         Produto6 pro1 = new Produto6("Pão de Queijo");
8         Produto6 pro2 = new Produto6("Pão Francês");
9         Produto6 pro3 = new Produto6("Pão Doce");
10        pro3.unidadeMedida = "Un";
11
12        pro1.adicionarEstoque(10);
13        pro2.adicionarEstoque(5);
14        pro3.adicionarEstoque(7);
15        System.out.println("-----");
16        System.out.println("Estoque: "+pro1.estoque);
17        System.out.println("Estoque: "+pro2.estoque);
18        System.out.println("Estoque: "+pro3.estoque);
19        System.out.println("Estoque Geral: "+pro1.estoqueGeral);
20        System.out.println("Estoque Geral: "+pro2.estoqueGeral);
21        System.out.println("Estoque Geral: "+pro3.estoqueGeral);
22        System.out.println("-----");
23        System.out.println("Estoque Geral: "+Produto6.obterEstoqueGeral());
24
25    }
26 }
```

```
-----
Estoque Geral: 0
-----
Estoque: 10
Estoque: 5
Estoque: 7
Estoque Geral: 22
Estoque Geral: 22
Estoque Geral: 22
-----
Estoque Geral: 22
```