

Orientação a Objetos 1

# Tipos de Variáveis

Prof. MSc. Vinícius Camargo Andrade

[vcandrade@utfpr.edu.br](mailto:vcandrade@utfpr.edu.br)

Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná

# Tipos de Variáveis

# Tipos de Variáveis

*Há tipos diferentes de variáveis. São elas:*

- *Variáveis de instância;*
- *Variáveis Imutáveis;*
- *Variáveis de classe;*
- *Variáveis locais;*
- *Parâmetros.*

# Variáveis de Instância

# Variáveis de Instância

*Nas variáveis de instância os seus valores são únicos a cada instância (objeto) de uma classe. Como por exemplo, o nome de uma Pessoa é independente do nome de outra.*

# Variáveis de Instância

*São variáveis declaradas **sem** a palavra-chave **static**.*

# Variáveis de Instância

```
public class Pessoa {  
    String nome;  
    int idade;  
    double peso;  
}
```

# Exercício 1



# Exercício 1

*Desenvolva um sistema para cadastrar pessoas. Sabe-se que uma **Pessoa** possui um **nome**, **data de nascimento**, **telefone**, **e-mail** e **estado civil**. Instancie três pessoas e atribua valores diferentes à cada atributo delas.*

Variáveis Imutáveis  
(constantes)

# Variáveis Imutáveis (Constantes)

*Atributos imutáveis são variáveis que ao receberem um determinado valor, este **não se alterará** posteriormente.*

# Variáveis Imutáveis (Constantes)

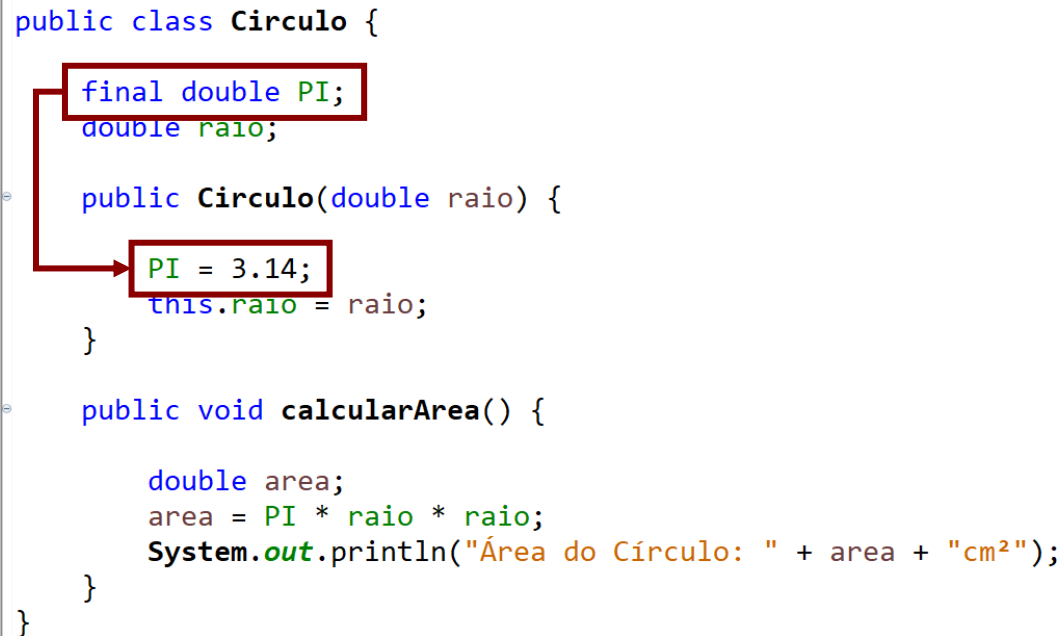
*Para definir um atributo imutável, utiliza-se a palavra-chave **final**.*

# Variáveis Imutáveis (Constantes)

```
public class Circulo {  
  
    final double PI;  
    double raio;  
  
    public Circulo(double raio) {  
  
        PI = 3.14;  
        this.raio = raio;  
    }  
  
    public void calcularArea() {  
  
        double area;  
        area = PI * raio * raio;  
        System.out.println("Área do Círculo: " + area + "cm²");  
    }  
}
```

# Variáveis Imutáveis (Constantes)

```
public class Circulo {  
    final double PI;  
    double raio;  
  
    public Circulo(double raio) {  
        PI = 3.14;  
        this.raio = raio;  
    }  
  
    public void calcularArea() {  
        double area;  
        area = PI * raio * raio;  
        System.out.println("Área do Círculo: " + area + "cm²");  
    }  
}
```



# Exercício 2

# Exercício 2

*Desenvolva um sistema para cadastrar pessoas. Sabe-se que uma **Pessoa** possui um **cpf**, **nome**, **data de nascimento**, **telefone**, **e-mail** e **estado civil**. Sendo que o **cpf**, uma vez definido, não pode ser alterado.*

*Instancie três pessoas e atribua valores diferentes à cada atributo delas.*



# Variáveis de Classe

# Variáveis de Classe

*É um atributo que pertence não ao objeto mas sim a própria classe.*

# Variáveis de Classe

É considerada uma *variável de classe* quando declarada com a palavra-chave *static*.

# Variáveis de Classe

*As variáveis de Classe são visíveis e podem ser acessadas por **todos os objetos** derivados de uma mesma classe.*

# Exemplo

Variáveis de Classe

Galinha
nome: String quantidadeOvos : Integer
botar( ) : void



Nome: Giselda



Nome: Matilde



Nome: Clotilde

O método *botar()* deve incrementar o atributo *quantidadeOvos* em 1 unidade, ou seja, cada galinha botará uma quantidade de ovos *independente* das outras galinhas da granja.

```

public class Galinha {

    String nome;
    int quantidadeOvos = 0;

    public void botar() {

        quantidadeOvos = quantidadeOvos + 1;
    }
}

```

```

run:
Giselda
3 ovos.
-----
Matilde
1 ovos.
-----
Clotilde
2 ovos.
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)

```

```

public class GalinhaTeste {
    public static void main(String[] args) {
        Galinha g1 = new Galinha();
        g1.nome = "Giselda";
        g1.botar();
        g1.botar();
        g1.botar();

        Galinha g2 = new Galinha();
        g2.nome = "Matilde";
        g2.botar();

        Galinha g3 = new Galinha();
        g3.nome = "Clotilde";
        g3.botar();
        g3.botar();

        System.out.println(g1.nome);
        System.out.println(g1.quantidadeOvos + " ovos.");
        System.out.println("-----");
        System.out.println(g2.nome);
        System.out.println(g2.quantidadeOvos + " ovos.");
        System.out.println("-----");
        System.out.println(g3.nome);
        System.out.println(g3.quantidadeOvos + " ovos.");

    }
}

```

# Problema

Exemplo Galinhas



# Problema

*Sabe-se a quantidade de ovos que cada galinha botou,  
mas como saber o total de ovos foram coletados  
nesta granja?*



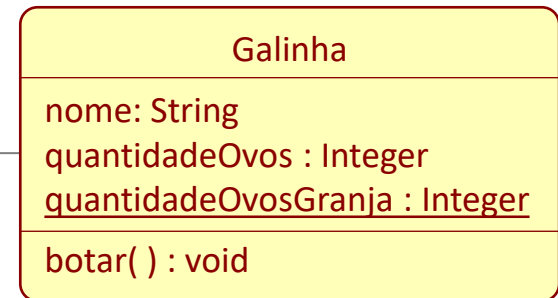
# Solução

*Declara-se um **atributo estático** na classe Galinha para fazer a contagem dos ovos, assim **todas as instâncias** (objetos) desta classe acessarão o atributo para incrementá-lo.*



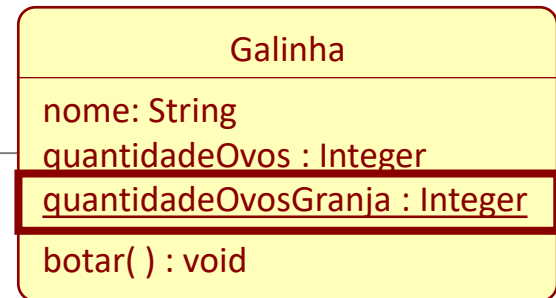
# Solução

```
public class Galinha {  
  
    String nome;  
    int quantidadeOvos = 0;  
    static int quantidadeOvosGranja = 0;  
  
    public void botar() {  
  
        quantidadeOvos = quantidadeOvos + 1;  
        quantidadeOvosGranja = quantidadeOvosGranja + 1;  
    }  
}
```



# Solução

```
public class Galinha {  
  
    String nome;  
    int quantidadeOvos = 0;  
    static int quantidadeOvosGranja = 0;  
  
    public void botar() {  
  
        quantidadeOvos = quantidadeOvos + 1;  
        quantidadeOvosGranja = quantidadeOvosGranja + 1;  
    }  
}
```



```

public class GalinhaTeste {
    public static void main(String[] args) {
        Galinha g1 = new Galinha();
        g1.nome = "Giselda";
        g1.botar();
        g1.botar();
        g1.botar();

        Galinha g2 = new Galinha();
        g2.nome = "Matilde";
        g2.botar();

        Galinha g3 = new Galinha();
        g3.nome = "Clotilde";
        g3.botar();
        g3.botar();

        System.out.println(g1.nome);
        System.out.println(g1.quantidadeOvos + " ovos.");
        System.out.println("-----");
        System.out.println(g2.nome);
        System.out.println(g2.quantidadeOvos + " ovos.");
        System.out.println("-----");
        System.out.println(g3.nome);
        System.out.println(g3.quantidadeOvos + " ovos.");
        System.out.println("-----");
        System.out.println("Total de ovos: " + Galinha.quantidadeOvosGranja);
    }
}

```

```

run:
Giselda
3 ovos.
-----
Matilde
1 ovos.
-----
Clotilde
2 ovos.
-----
Total de ovos: 6
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)

```

# Exercício 3

# Exercício 3

*Um conta bancária é identificada pelo seu número da conta, nome do titular e saldo.*

*Sabendo que o número de conta não pode ter o mesmo número de outra, desenvolva um sistema que, ao instanciar uma nova conta, verifique quantas contas já existem e atribua o número seguinte a nova conta.*

# Variáveis Locais



# Variáveis Locais

*Assim como um objeto armazena seus estados em variáveis de instância, **um método** frequentemente armazenará seus estados temporários em **variáveis locais**.*

# Variáveis Locais

*Não há nenhuma palavra-chave especial para designar uma variável como local, o que determina é inteiramente a **localização** na qual esta variável é declarada — a qual é entre a chave de abertura e de fechamento de um método.*

# Variáveis Locais

*Variáveis locais são **somente visíveis** aos métodos nos quais são declaradas; elas **não são acessíveis** do resto da classe.*

# Variáveis Locais

```
public class Calculadora {  
    public void somar() {  
        int x = 2;  
        int y = 5;  
        int soma;  
        soma = x + y;  
    }  
}
```

Parâmetros

# Parâmetros

*As variáveis que receberão as informações enviadas a um método são chamadas de parâmetros.*

# Parâmetros

*A função deve declarar essas variáveis entre parênteses.*

- *Importante: parâmetros são sempre classificados como "variáveis" não "campos".*

# Parâmetros

```
public class Calculadora {  
    public void somar(int x, int y) {  
        int soma;  
        soma = x + y;  
    }  
}
```



# Exercício 4

# Exercício 4

Desenvolva um sistema para controlar um carro. Sabe-se que o carro é identificado pelo seu *modelo*, *marca*, *cor*, *ano*, *renavam*, *chassi* e *placa*. Além disso, o carro pode *ligar*, *desligar*, *acelerar* e *frear*. Ao invocar os métodos *acelerar()* e/ou *frear()* devem receber por parâmetro o valor da velocidade que deve ser somada ou subtraída da velocidade atual do veículo.

- *Exemplo:* o carro está a 80 km/h e o método *acelerar* é invocado sendo passado por parâmetro o valor 35. O método deve somar a velocidade atual (80) a velocidade recebida por parâmetro (35), atualizando a velocidade do veículo para 115 km/h.