

ESTRITAMENTE CONFIDENCIAL

**ELABORATA**  
INFORMÁTICA

**LINGUAGEM JAVA**





# Orientação a Objetos

## Classes e Atributos

# Objetos

- A base de uma linguagem orientada a objetos, como o Java, é o objeto. Todo objeto possui uma estrutura que é composta por um identificador, características e comportamentos.

# Estrutura objeto 'carro'

- Por exemplo, o objeto 'carro' possui a seguinte estrutura:

*Identificador:* Carro

*Características:* Marca, Modelo, Cor

*Comportamentos:* Ligar, Desligar, Acelerar, Frear, ...

# Objeto 'carro' no programa

- Para que o objeto 'carro' possa ser utilizado num programa de computador, por exemplo num programa para gerenciar uma locadora de veículos, é necessário abstrair a estrutura do objeto. Essa abstração é feita por meio de uma classe.

# Classe

- Uma classe é a abstração da estrutura de um determinado objeto para que o objeto possa ser utilizado no programa.

# Objeto para Classe

- Quando um determinado objeto for abstraído para uma classe, o identificador do objeto será o nome da classe, as características do objeto serão os atributos da classe e os comportamentos serão os métodos da classe.

# Classe 'carro'

```
public class Carro
{
    lista-de-atributos
    lista-de-métodos
}
```



# Atributos

- Os **atributos** de uma classe são as características do objeto que a classe está abstraindo. Os atributos são utilizados para armazenamento de dados. Por exemplo, para guardar a cor 'preta' de um veículo, a classe Carro precisa ter um atributo chamado 'cor'.

# Estrutura de um atributo

<modificador-de-acesso> <tipo-de-dados> <identificador>;

# Modificador de acesso

- O **modificador de acesso** especifica como o atributo pode ser acessado dentro e fora da classe. Um atributo é acessado dentro da classe quando ele é mencionado dentro de um método. Acessar um atributo fora da classe significa que o atributo será mencionado em métodos de outras classes.

## Modificador de acesso (cont.)

■ Os valores possíveis para um modificador de acesso de atributos são:

■ *public*

■ *private*

■ *protected.*

# Tipo de dado

- O **tipo de dado** é referente ao dado que será armazenado no atributo. Os valores possíveis são: qualquer tipo de dado primitivo da Linguagem Java ou uma outra classe.

# Identificador

- O **idendificador** de atributo deve ser definido seguindo as mesmas regras utilizadas para definição de identificadores de variáveis.

# Exemplo de Atributos

```
public class Carro
{
    public String marca;
    public String modelo;
    public String cor;
}
```

# Utilizando uma classe com atributos

- Para utilizar uma classe é necessário declarar um objeto sendo do tipo da classe e posteriormente criá-lo usando o operador **new**.



# Exemplo

```
// Declarando o objeto 'c1' sendo do tipo 'Carro'  
Carro c1;
```

```
// Criando o objeto 'c1'  
c1 = new Carro();
```

# Atribuindo valores aos atributos

- Os atributos são definidos em uma classe para que dados referentes ao objeto que a classe esta abstraindo possam ser armazenados para futuramente serem utilizados. Um dado é atribuído a um atributo por meio do operador de atribuição.

# Exemplo

```
c1.marca = "Ford";  
c1.modelo = "Mustang";  
c1.cor = "Branco"
```

# Efetuando cálculos

- Além de armazenarem dados, os atributos podem ser utilizados para realização de cálculos. Por exemplo, a classe Carro poderia receber mais dois atributos, um chamado 'capacidadeTanqueEmLitros' e um outro chamado 'quilometrosPorLitro'. Com esses dois novos atributos pode-se calcular a autonomia de um determinado carro.

# Exemplo

```
public class Carro
{
    public String marca;
    public String modelo;
    public String cor;
    public float capacidadeTanqueEmLitros;
    public float quilometrosPorLitro;
}
```

## Exemplo (cont.)

```
Carro c1 = new Carro();  
c1.marca = "Ford";  
c1.modelo = "Mustang";  
c1.ano = 2016;  
c1.capacidadeTanqueEmLitros = 56;  
c1.quilometrosPorLitro = 8;  
  
float autonomiaEmQuilometros =  
    c1.capacidadeTanqueEmLitros * c1.quilometrosPorLitro;  
  
System.out.println("A autonomia é: " + autonomiaEmQuilometros);
```

# Exercício 01

- Criar uma classe chamada Funcionario com os atributos nome, cargo e salário.

## Instruções:

- Criar um projeto chamado Funcionario
- Criar uma classe chamada Funcionario
- Adicionar os atributos nome, cargo e salário na classe Funcionario
- Criar uma classe principal chamada CalcularSalario, criar um objeto do tipo Funcionario, preencher os dados do Funcionario e mostrar na tela o valor do salário para uma jornada de trabalho de 40 horas, levando em consideração que o valor hora é R\$ 14,50.

## Exercício 02

- Criar uma classe chamada Produto com os atributos descrição e preço.

### Instruções:

- Criar um projeto chamado Produto
- Criar uma classe chamada Produto
- Adicionar os atributos descricao e preco na classe Produto
- Criar uma classe principal chamada TabelaPrecos, criar um objeto do tipo Produto, preencher os dados do Produto e mostrar na tela os preços para distribuidor, representante e consumidor.
- O preço do distribuidor é o preço do produto + 3%
- O preço do representante é o preço do produto + 4,5%
- O preço do consumidor é o preço do produto + 5,5%



# OBRIGADO!

ESTRITAMENTE CONFIDENCIAL





[www.elaborata.com.br](http://www.elaborata.com.br)

### **Horário de Atendimento Comercial**

Segunda à sexta – das 9:00h às 19:30h e  
Sábado - das 8:00h às 15:00h.

Rua Monsenhor Celso, 256 - 1º Andar  
Centro - Curitiba - PR

**41.3324.0015**

 **41.99828.2468**

[cursos@elaborata.com.br](mailto:cursos@elaborata.com.br)

