

# Programação Orientada a Objetos

## ITE - 003

*profº Mauricio Conceição Mario*

# Superclasse Automovel1

```
public class Automovel1 {  
  
    //variáveis de instância = atributos  
    private int ano;  
    protected String marca;  
    private String modelo;  
    private String cor;  
    private double preco;  
  
    //método construtor  
    Automovel1( )  
    {  
        ano = 0;  
        marca = "";  
        modelo = "";  
        cor = "";  
        preco = 0.0;  
    }  
  
    public void setano ( int ano){  
        this.ano = ano;  
    }  
  
    public void setmarca ( String marca){  
        this.marca = marca;  
    }  
  
    public void setmodelo ( String modelo){  
        this.modelo = modelo;  
    }  
}
```

# Superclasse Automovel1

```
public void setcor ( String cor){  
    this.cor = cor;  
}  
  
public void setpreco ( double preco){  
    this.preco = preco;  
}  
  
public int getano ( ){  
    return ano;  
}  
  
public String getmarca ( ){  
    return marca;  
}  
  
public String getmodelo( ){  
    return modelo;  
}  
  
public String getcor( ){  
    return cor;  
}  
  
public double getpreco( ){  
    return preco;  
}
```

# Superclasse Automovel1

**/\* dois métodos mostracarro( ) com diferenças em relação às funcionalidades e aos parâmetros que recebem\*/**

```
public void mostracarro ()
{
    System.out.println("carro marca: " + "\t" + marca + "\n" + "modelo: " + "\t" +
        modelo + "\n" + "ano: " + "\t" + ano + "\n" + "cor: " + "\t" + cor + "\n" + "preco: R$ " + "\t" + preco);
}

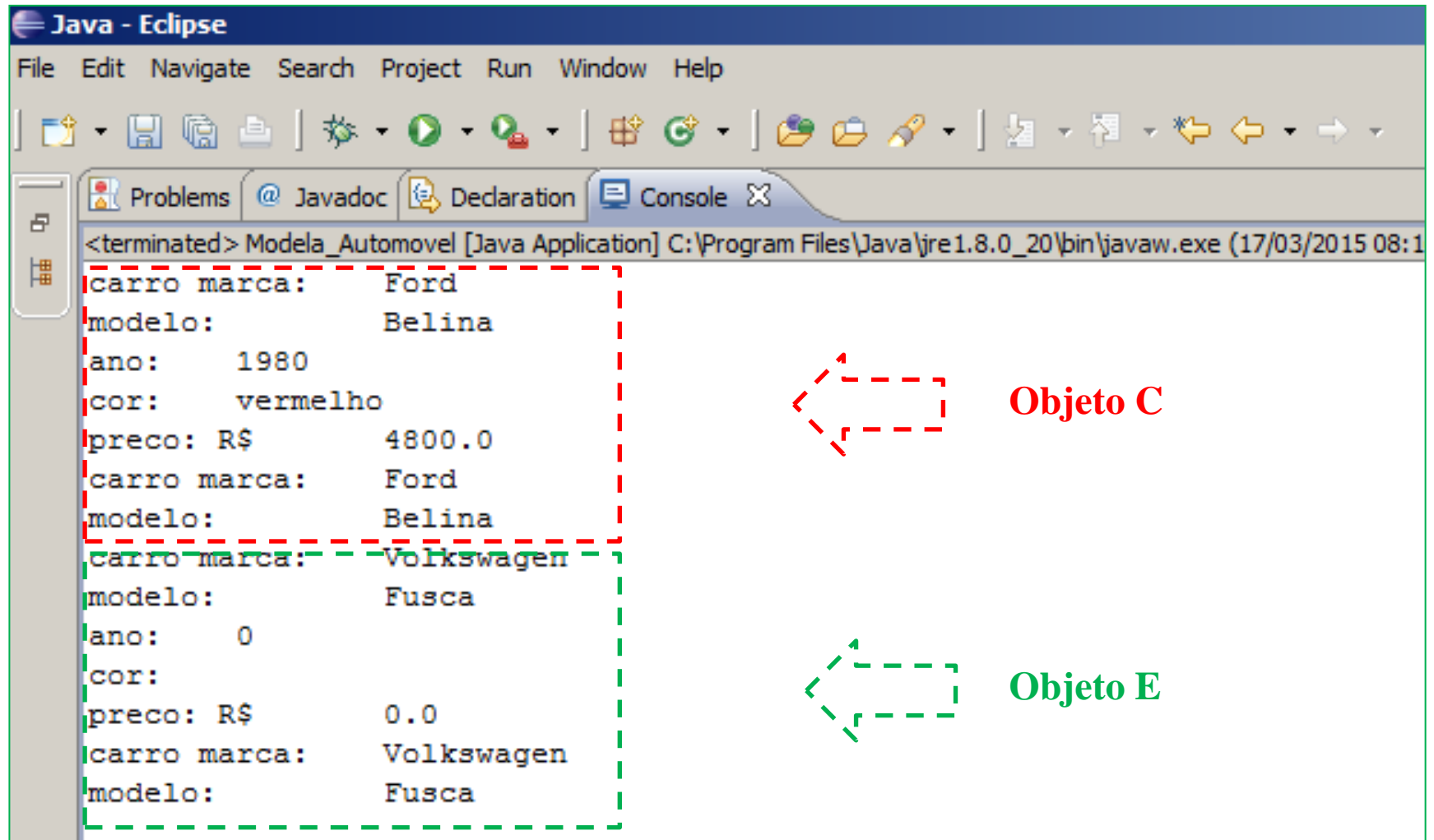
public void mostracarro (String mc, String md )
{
    System.out.println("carro marca: " + "\t" + mc + "\n" + "modelo: " + "\t" + md);
}

}
```

# Classe Modela\_Automovel

```
public class Modela_Automovel {  
  
    public static void main (String args[]){  
  
        //objetos C e E  
        Automovel1 C = new Automovel1();  
        Automovel1 E = new Automovel1();  
  
        C.setano(1980);  
        C.setmarca("Ford");  
        C.setModelo("Belina");  
        C.setcor("vermelho");  
        C.setpreco(4800.00);  
  
        Modela(C);  
  
        E.setmarca("Volkswagen");  
        E.setModelo("Fusca");  
  
        Modela(E);  
  
    }  
    //método (interno à classe) que recebe como parâmetro um objeto do tipo Automovel1  
    public static void Modela(Automovel1 D){  
        //sobrecarga do método mostracarro()  
        D.mostracarro();  
        D.mostracarro(D.getmarca(), D.getModelo());  
    }  
}
```

# Execução da classe Modela\_Automovel



The screenshot shows the Eclipse IDE interface with the console window open. The console title is "<terminated> Modela\_Automovel [Java Application] C:\Program Files\Java\jre1.8.0\_20\bin\javaw.exe (17/03/2015 08:1". The console output displays the attributes of two car objects, C and E, with red and green dashed boxes highlighting their respective data.

**Objeto C (Red dashed box):**

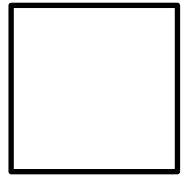
```
carro marca: Ford
modelo: Belina
ano: 1980
cor: vermelho
preco: R$ 4800.0
```

**Objeto E (Green dashed box):**

```
carro marca: Volkswagen
modelo: Fusca
ano: 0
cor: 
preco: R$ 0.0
carro marca: Volkswagen
modelo: Fusca
```

# Exercício:

8. As áreas de um quadrado, retângulo e um cubo podem ser calculadas, respectivamente, conforme abaixo:



a

$$\text{área} = a * a$$

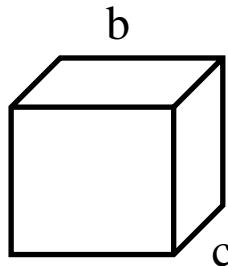
a



a

$$\text{área} = a * b$$

b



b

a

$$\text{área} = a * b * c$$

c

Utilizando o conceito de sobrecarga de métodos, construir uma classe que contenha os métodos `area()` que possam calcular as áreas do quadrado, retângulo e cubo.