



Compartilhar o seu link com: luciorocha@professores.utfpr.edu.br

Pedro Reis e Thiago Tieghi: **Cópia de Aula 4 - Exercícios propostos**
Fernando Rafael: Aula 4 - Exercícios propostos
Raphael Uematsu: Cópia de Aula 4 - Exercícios propostos
Matheus Mazali Maeda Persis: Aula 4 - Exercícios propostos
Alexandre Aparecido da Silva/ Luis : Cópia de Aula 4 - Exercícios propostos
Vitor Luiz de Castro Viana: Cópia de Aula 4 - Exercícios propostos
João Pedro de Paula:< Cópia de Aula 4 - Exercícios propostos >
Vitor Hugo Leite Arruda de Oliveira: < Aula 04 - Exercícios >
Séfora< Cópia de Aula 4 - Exercícios propostos >
Angélica < Cópia Aula 4 - Exercícios propostos >
Matheus Hirata VAnzela< Cópia de Aula 4 - Exercícios propostos >
João Vitor N. Yoshida < Cópia de Aula 4 - Exercícios propostos >
Filipe Augusto Parreira Almeida Cópia de Aula 4 - Exercícios propostos
Rafael Kendy Naramoto Lopes < Cópia de Aula 4 - Exercícios propostos >
Marcos Tadao Shoji & Plinio Cópia de Aula 4 - Exercícios propostos
Rodrigo Leandro Benedito: Cópia de Aula 4 - Exercícios propostos
Felipe Antonio Magro: Cópia de Aula 4 - Exercícios propostos
Gustavo Nunes : Cópia de Aula 4 - Exercícios propostos
Guilherme Ramalho: Cópia de Aula 4 - Exercícios propostos
Alexandre Olah : Aula 4 - Exercícios propostos
Thales Alves: Aula 4 - Exercícios propostos
João Pedro Cavani Meireles: < Cópia de Aula 4 - Exercícios propostos >
Deivid da Silva < Aula 4 - Exercícios propostos >
Mabyly< aula 4 >
Daniel Martins/Gabriel Takeshi< Cópia de Aula 4 - Exercícios propostos >
Thiago Cristovão de Souza: < Cópia de Aula 4 - Exercícios propostos >
Isabella Melo Almeida: Aula 4
Felipe F B Lorusso <[Cópia de Aula 4 - Exercícios propostos](#)>
Leitura recomendada: <https://docs.oracle.com/javase/tutorial/>
Lucas Viana e Victor Ramos: Cópia de Aula 4 - Exercícios propostos
Julio Farias: Cópia de Aula 4 - Exercícios propostos
Roberto Furlani Neto: Cópia de Aula 4 - Exercícios propostos

//Revisao

//

}

```
public class Calculadora {

    private String marca;

    //Metodos assessores
    public String getMarca(){
        return this.marca;
    }
    //Metodos mutadores
    public void setMarca(String marca){
        this.marca = marca;
    }

}

public class Principal {

    public Principal() {

        Calculadora nokia = new Calculadora();
        nokia.setMarca("Nokia");
        System.out.println( nokia.getMarca() );

    }

    public static void main(String [ ] args){

        Principal obj = new Principal();

    }

}
```

Exercícios propostos

- 1) (Online) Acesse o link: <https://codeboard.io/projects/341626>

- a) Receba 2 (dois) números decimais informados pelo usuário e mostre a soma deles.

```
public class Principal {

    public Principal(){

        Scanner entrada = new Scanner(System.in);
        System.out.println("Numero1:");
        float numero1 = entrada.nextFloat();

        System.out.println("Numero2:");
        float numero2 = entrada.nextFloat();

        System.out.println("Soma:" + (numero1+numero2));
    }

    public static void main(String[] args) {
        new Principal();
    }
}
```

- 2) (NetBeans) Exemplo com Elementos Gráficos básicos: Utilize a classe JOptionPane. Receba 2 (dois) números decimais informados pelo usuário e mostre a soma deles.

```
package aula4;

import javax.swing.JOptionPane;

public class Principal1 {

    public Principal1() {

        String n1 = JOptionPane.showInputDialog(null,
            "Digite o numero1");
        float numero1 = Float.parseFloat(n1);

        String n2 = JOptionPane.showInputDialog(null,
            "Digite o numero2");
        float numero2 = Float.parseFloat(n2);

        JOptionPane.showMessageDialog(null,
            (numero1+numero2),"Resultado",
            JOptionPane.WARNING_MESSAGE);
    }
}
```

}

```
}

public static void main(String [ ] args){

    Principal1 obj = new Principal1();
    System.exit(0);
}

}
```

3) (NetBeans) Elabore um programa orientado a objetos a partir do seguinte enunciado: uma floricultura possui flores de diversos tipos: rosa, margarida e violeta. O programa deve ser capaz de cadastrar o tipo de flor e modificar a cor, o nome e o preço de cada flor.

Nomes (substantivo/sujeito): Floricultura

Adjetivos (atributos): tipo, cor, nome, preco

Verbos (comportamento/método): possuir (get), cadastrar (set), modificar

```
public class Floricultura {

    private String tipo;
    private String cor;
    private String nome;
    private float preco;

    public String getTipo() { return tipo; }
    public String getCor() { return cor; }
    public String getNome() { return nome; }
    public float getPreco() { return preco; }

    //Metodos mutadores
    public void setTipo( String tipo) { this.tipo = tipo; }
    public void setCor( String cor) { this.cor = cor; }
    public void setNome( String nome) { this.nome = nome; }
    public void setPreco( float preco) { this.preco = preco; }
}

public class Principal {
```

```

        public static void main(String [ ] args){
            Floricultura floricultura = new Floricultura();
            floricultura.setTipo("rosa");
            floricultura.setCor("vermelha");
            imprimir();
        }

        public static void imprimir(){
            System.out.println("Fora do metodo main");
        }
    }

```

4) (NetBeans) Elabore um programa orientado a objetos a partir do seguinte enunciado: uma transportadora possui os seguintes tipos de veículos: caminhão e van. Todos os veículos possuem preço de frete e capacidade. O programa deve ser capaz de cadastrar o veículo e inserir uma carga para o transporte.

Nomes (Classes): transportadora, caminhao, van

Adjetivos (potenciais atributos/características): frete, capacidade, carga

Verbos (potenciais comportamentos / métodos): possuir, cadastrar, inserir

```

public class Transportadora {

    //Variaveis de instancia
    private Caminhao caminhao;
    private Van van;

    public Caminhao getCaminhao(){ return caminhao; }
    public void setCaminhao( Caminhao caminhao ) { this.caminhao = caminhao; }

    public Van getVan(){ return van; }
    public void setVan(Van van){ this.van = van; }
}

public class Caminhao {
    //Variaveis de instancia
    private float frete;

```

}

```
private float capacidade;
//
public float getFrete(){ return this.frete; }
public void setFrete(float frete){ this.frete = frete; }

public float getCapacidade(){ return this.capacidade; }
public void setCapacidade(float capacidade){ this.capacidade = capacidade; }
}
public class Van {
    //Variaveis de instancia
    private float frete;
    private float capacidade;
}

public class Principal {

    public static void main( String [ ] args ){

        Transportadora transportadora = new Transportadora();

        Caminhao caminhao = new Caminhao();
        caminhao.setFrete(100.0f);
        caminhao.setCapacidade(200.0f);

        transportadora.setCaminhao( caminhao );

    }

}
```

5) (NetBeans) Elabore um programa orientado a objetos a partir do seguinte enunciado: uma loja de calçados vende diversos tipos de calçados. Cada calçado possui uma marca, um tamanho e um valor. O programa deve ser capaz de cadastrar os sapatos.

Nomes (Classes):

Adjetivos: (Atributos/variáveis de instância):

Verbos: (Comportamentos / métodos):

--

6) (NetBeans) Elabore um programa orientado a objetos a partir do seguinte enunciado: uma propriedade rural deve ser capaz de cadastrar sacas de café. O valor a ser pago por saca é de acordo com a seguinte tabela:

Peso	Valor
Até 30 quilos	R\$ 50,00
De 31 quilos até 60 quilos	R\$ 100,00
De 61 quilos até 70 quilos	R\$ 200,00

Nomes:

Adjetivos:

Verbos:

7) (Opcional) Mais exercícios em: <https://github.com/poco4a/aula4>