

Programação Orientada a Objeto

Introdução a OO

Paradigma Orientado a Objeto

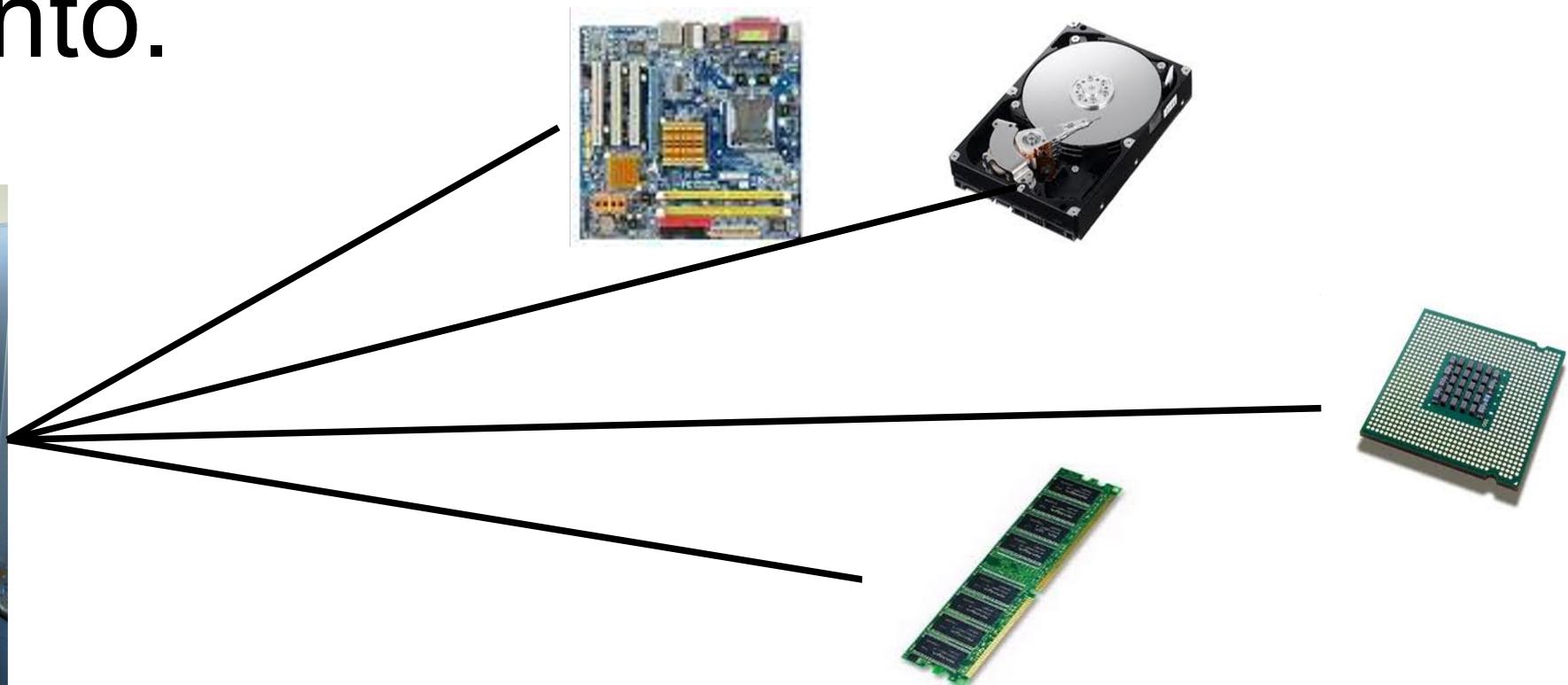
Comunidade de agentes (objetos) que interagem entre si

Cada agente tem um papel a cumprir, oferece um serviço ou realiza uma ação que é usada por outros objetos.

Objetos são definidos através de modelos (classe).

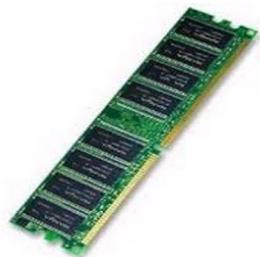
Paradigma Orientado a Objeto

Em um domínio (sistema) é comum existirem elementos importantes para seu funcionamento.



Objetos

Objetos possuem característica e realizam funções



Memoria

Macar: Kingston
Capacidade: 4GB
Velocidade: 1600Mhz

Armazena dados
Apaga dados
Ler dados



Barramento

Velocidade: 1600Mhz
Quantidade Entra: 3
Levar dados
Levar instruções



Processador

Macar: Intel
Model: i3
Velocidade: 2,5 Ghz

Somar
Subtrair
Multiplicar
Dividir

Objetos

Conta bancaria

ContaBancaria

Característica
(Atributos)

Funções (Métodos)

Objetos

Conta bancaria

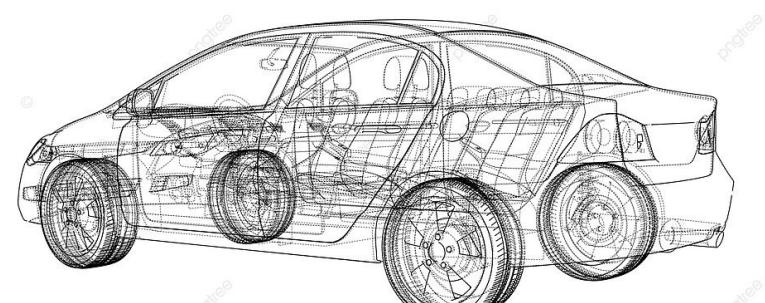
ContaBancaria

Número
Saldo

Depositar
Sacar

Classes

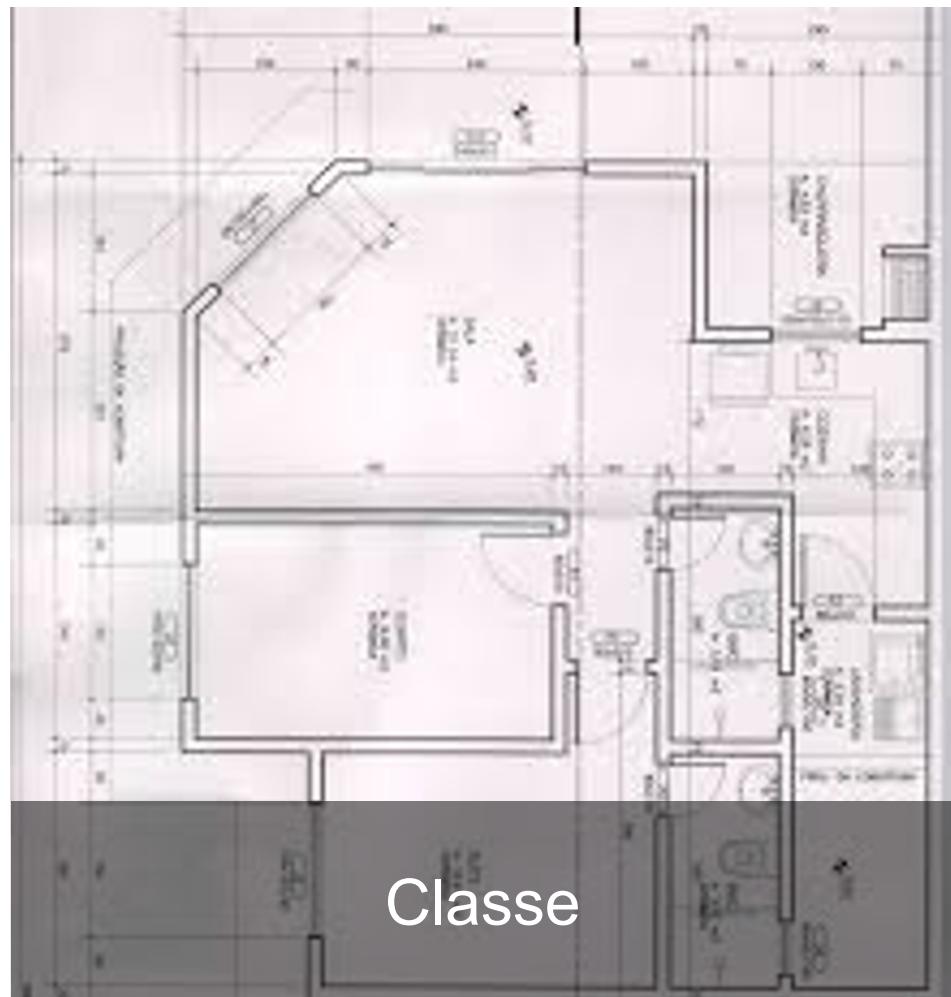
São **modelos** que indicam o funcionamento de cada objeto.



Classe



Objeto



Classe X Objeto

Java

```
public class Conta {  
    int numero = 0;  
    double saldo = 0.0 ;  
  
    public void depositar(double valor){  
        saldo = saldo + valor;  
    }  
  
    public void sacar(double valor){  
        saldo = saldo - valor;  
    }  
}
```

Classe

```
public class Primeiro {  
    public static void main(String[] args) {  
  
        Conta minhaConta = new Conta();  
        minhaConta.numero = 1;  
        minhaConta.depositar(100);  
        minhaConta.sacar(30.40);  
        System.out.println(minhaConta.saldo);  
    }  
}
```

Objeto

Criando classe

```
public class NomeClasse {  
    //Atributos  
  
    //Métodos  
}
```

Atributos

```
public class Conta {  
    int numero = 0;  
    double saldo = 0.0 ;  
  
    public void depositar(double valor){  
        saldo = saldo + valor;  
    }  
  
    public void sacar(double valor){  
        saldo = saldo - valor;  
    }  
}
```

Métodos

Classe: Atributos (Variáveis de instância)

+ .
o

Armazena os dados para os objetos

modificador Tipo nome;

```
public class Pessoa {  
    public String nome;  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Pessoa p1 = new Pessoa();  
        Pessoa p2 = new Pessoa();  
        p1.nome = "José";  
        p2.nome = "Maria";  
        JOptionPane.showMessageDialog(null, p1.nome);  
        JOptionPane.showMessageDialog(null, p2.nome);  
    }  
}
```

Atividade



Crie uma classe Aluno que tenha nome e média



Crie uma classe Lâmpada que tenha um estado (acesa ou apagada).



Crie uma classe Retângulo que tenha base e altura



Crie uma classe Funcionário que tenha nome e salário.

Respostas

```
public class Aluno {  
    public String nome;  
    public double media;  
}
```

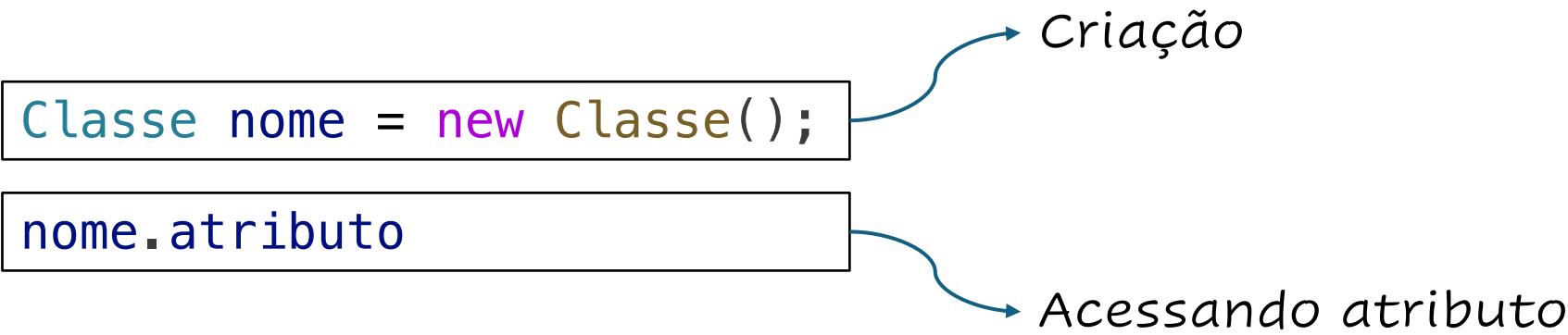
```
public class Funcionario {  
    public String nome;  
    public double salario;  
}
```

```
public class Lampada {  
    private boolean acesa;  
}
```

```
public class Retangulo {  
    public double base;  
    public double altura;  
}
```

```
public class Retangulo {  
    public double base;  
    public double altura;  
}
```

Objetos



```
public class Aluno {  
    public String nome;  
    public double media;  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Aluno a1 = new Aluno();  
        a1.nome = "Ayrton Sena";  
        a1.media = 9.8;  
        System.out.println(a1.nome);  
        System.out.println(a1.media);  
    }  
}
```

Atividade



Crie uma objeto do tipo Aluno e insira valores para seu **nome** e **media**.



Crie uma Lâmpada e coloque seu estado como acesa.



Crie objeto do tipo Retângulo atribua e exiba valores para sua base e altura.



Crie um Funcionário e atribua valores para seu nome e salario.

Respostas

```
public class Lampada {  
    private boolean acesa;  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Lampada lamp1 = new Lampada();  
        Lampada lamp2 = new Lampada();  
        lamp1.acesa = true;  
        lamp2.acesa = false;  
        System.out.println(lamp1.acesa);  
        System.out.println(lamp2.acesa);  
    }  
}
```

```
public class Funcionario {  
    public String nome;  
    public double salario;  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Funcionario f1 = new Funcionario()  
        f1.nome = "Jose";  
        f1.salario = 1000;  
        System.out.println(f1.nome);  
        System.out.println(f1.salario);  
    }  
}
```

Classe: métodos

```
public class Conta {  
    modificador tipo nome(tipo param){  
        <comandos>  
    }  
}
```

```
public class Conta {  
  
    public double saldo;  
  
    public void depositar(double valor){  
        saldo = saldo+ valor;  
    }  
  
    public void sacar(double valor){  
        saldo = saldo - valor;  
    }  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
  
        Conta minhaConta = new Conta();  
        minhaConta.depositar(100);  
        minhaConta.sacar(30.40);  
        System.out.println(minhaConta.saldo);  
    }  
}
```

Classe: métodos

```
public class Conta {  
    modificador tipo nome(tipo param){  
        <comandos>  
    }  
}
```

```
public class Retangulo {  
    public double base;  
    public double altura;  
  
    public double calcularArea(){  
        return base * altura;  
    }  
}
```

```
public class Primeiro {  
  
    public static void main(String[] args) {  
        Retangulo r1 = new Retangulo();  
        r1.altura = 10;  
        r1.base = 20;  
        System.out.println(r1.calcularArea());  
    }  
}
```

Atividade

Crie uma classe Circulo, que contenha como atributo o raio do círculo e como método uma operação que retorna a área desse círculo. Por fim, crie um objeto do tipo Circulo, atribua o seu raio e calcule a sua área.

Resposta

```
public class Circulo {  
    public double raio;  
  
    public double calcularArea(){  
        return 3.14 * raio * raio;  
    }  
}  
  
public class Primeiro {  
  
    public static void main(String[] args) {  
        Circulo meuCirculo = new Circulo();  
        meuCirculo.raio = 4;  
        System.out.println(meuCirculo.calcularArea());  
    }  
}
```

Atividade

- Crie uma classe Empregado, que contenha, como atributo, a quantidade de horas trabalhadas por um empregado e, como método, uma operação que adiciona horas ao empregado.
 - O método recebe uma quantidade de horas e adiciona ao total de horas do empregado.

Empregado

+ hrTrabalhadas

+ adicionarHoras(horas)

Respostas

```
public class Empregado {  
    public double hrTabalhadas;  
  
    public void adicionarHoras(double horas){  
        hrTabalhadas = hrTabalhadas + horas;  
    }  
}
```

```
public class Primeiro {  
  
    public static void main(String[] args) {  
        Empregado jose = new Empregado();  
        jose.adicionarHoras(10);  
        jose.adicionarHoras(5);  
        System.out.println(jose.hrTabalhadas);  
    }  
}
```

Atividade

- Adicione a classe empregado um método que calcule e retorne o salário do empregado, sendo este salário o quantidade de horas vezes o valor da hora, que é de R\$ 70.50.

Empregado

+ hrTrabalhadas
+ adicionarHoras(horas)
+ calcularSalario(): double

Resposta

```
public class Empregado {  
    public double hrTabalhadas;  
  
    public void adicionarHoras(double horas){  
        hrTabalhadas = hrTabalhadas + horas;  
    }  
  
    public double calcularSalario(){  
        return hrTabalhadas * 70.50;  
    }  
}
```

```
public class Primeiro {  
  
    public static void main(String[] args) {  
        Empregado jose = new Empregado();  
        jose.adicionarHoras(10);  
        jose.adicionarHoras(5);  
        System.out.println(jose.calcularSalario());  
    }  
}
```

Atividade

Na classe Lampada crie um método acender e outro apagar. O método acender não recebe nenhum valor e colocar o estado da lâmpada como apagada(**false**). O método acender, também não recebe nenhum valor e coloca o estado da lâmpada como acesa(**true**).

Classe antiga

```
public class Lampada {  
    private boolean acesa;  
}
```

Classe nova

Lampada

- + acesa:boolean
- + acender()
- + apagar()

Resposta

```
public class Lampada {  
    public boolean acesa;  
  
    public void acender() {  
        acesa = true;  
    }  
  
    public void apagar() {  
        acesa = false;  
    }  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Lampada l = new Lampada();  
        l.acender();  
        l.apagar();  
    }  
}
```

Atividade

- Crie uma classe Carro que possua o atributo velocidade. Além disso, esta classe deve ter o método acelerar e frear, que incrementam e decrementam a velocidade em uma unidade. Por fim, crie um carro acelere e freie algumas vezes, e exiba a velocidade final

Carro

+ velocidade:int

+ acelerar()

+ frear()

Resposta

```
public class Carro {  
    public int velocidade;  
  
    public void acelerar(){  
        velocidade ++;  
    }  
  
    public void frear(){  
        velocidade --;  
    }  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Carro gol = new Carro();  
        gol.acelerar();  
        gol.acelerar();  
        gol.acelerar();  
        gol.frear();  
        System.out.println(gol.velocidade);  
    }  
}
```

Atividade

- Crie a classe Conta, que representa uma conta bancária. Esta classe possui o atributo saldo, que armazena o saldo atual da conta, e os métodos sacar e depositar, que realizam saques e depósitos na conta, alterando o valor do saldo. Por fim, crie uma conta, realize saques e depósitos e exiba o valor final.

Conta

+ saldo:double

+ sacar(valor)

+ depositar(valor)

Resposta

```
public class Conta {  
  
    public double saldo;  
  
    public void depositar(double valor){  
        saldo = saldo + valor;  
    }  
  
    public void sacar(double valor){  
        saldo = saldo - valor;  
    }  
}
```

```
public class Primeiro {  
    public static void main(String[] args) {  
        Conta c = new Conta();  
        c.depositar(100);  
        c.depositar(100);  
        c.sacar(150);  
        System.out.println(c.saldo);  
    }  
}
```