

POO – Programação Orientada a Objetos

**Classes e Objetos**

# + Roteiro

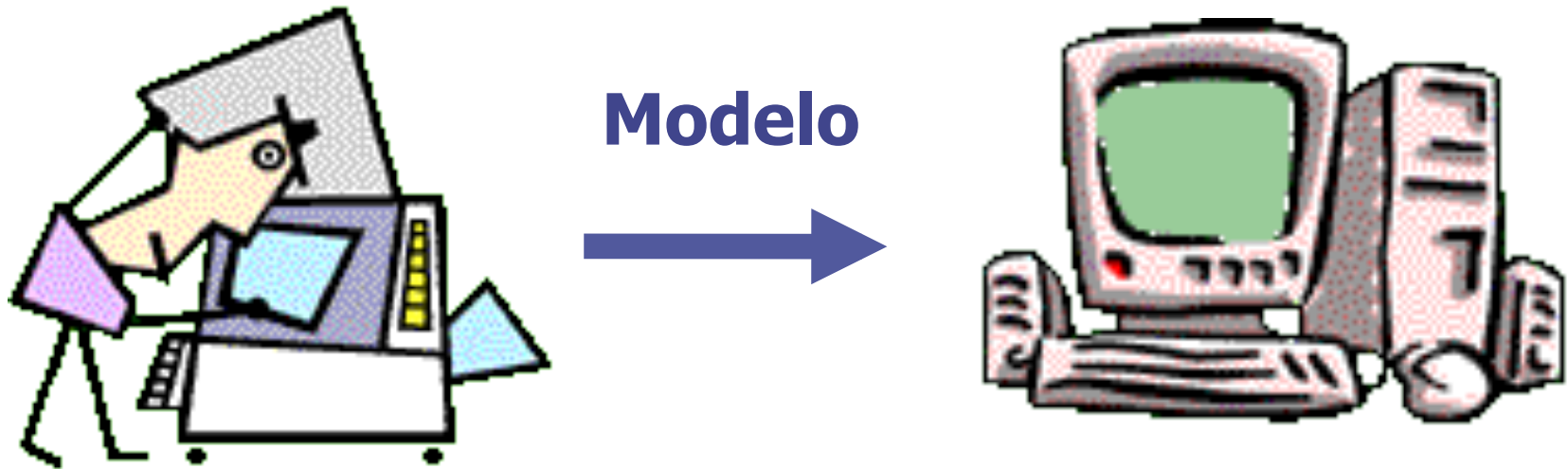
- Conceitos
  - OO no desenvolvimento de sistemas
- Objetos
  - Atributos
  - Métodos
- Classe
  - Definição do objeto
- Exemplos e exercícios

# + Desenvolvimento de programas

- Análise
  - Define **o que** deve ser feito
- Projeto
  - Define **como** deve ser feito
- Implementação
  - Escreve em uma linguagem de programação (**faz**)

# + Programa

- Operações sobre dados
- Capacidade computacional aumenta
  - Aumenta a complexidade



# + Por que OO?

- Sistemas são complexos
  - Ciclo de vida longo.
  - Muito difícil um único indivíduo entender todo o sistema.
  - Podemos **gerenciar** a complexidade, não podemos eliminá-la.
- O “domínio do problema” é complexo
  - Necessário pensar na evolução do sistema

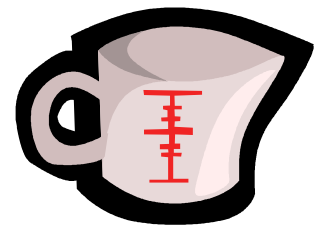
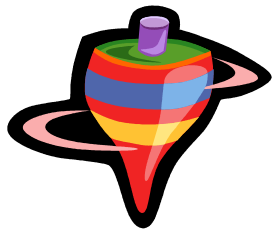
# + Por que OO?

- Modelagem OO
  - O ser humano conhece o mundo e gerencia sua complexidade através de objetos
  - É como desenvolvemos nossa cognição.
  - Desenvolvemos o conceito de **OBJETO**
- Exemplos de objetos
  - bola, carro, camisa, luz, casa, calça, música, conta bancária, poema, etc



## + Objetos

- “um objeto representa um item **identificável**, uma unidade, ou entidade, individual, seja **real** ou **abstrato**, com uma regra bem definida”



**OBJETO = DADOS + OPERAÇÕES**

# + Objetos

- Características
  - Dados representam características
    - São chamados **atributos**
    - São as **variáveis** do objeto
- Comportamento
  - Operações definem comportamento
    - São os **métodos** de um objeto
    - São as **funções** que são executadas por um objeto



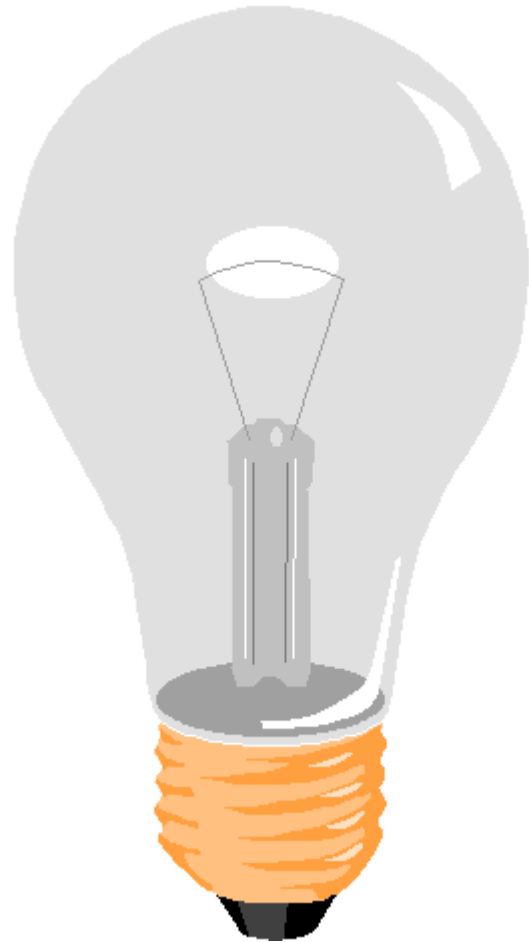
# + Objetos

- Objetos Possuem:
  - **Estado**
    - Representado pelos valores dos **atributos** de um objeto
  - **Comportamento**
    - Definido pelo conjunto de **métodos** do objeto
    - Estado representa o resultado cumulativo de seu comportamento
  - **Identidade**
    - Forma que conhecemos o objeto, é a **referência** ao objeto

# + Classe e objetos

- Classe
  - Classe é onde conceituamos o objeto
  - É a **essência** do objeto
  - Serve como **molde** do objeto
  - **Especifica** o objeto
  - **Define** os atributos e métodos
- Objeto é uma **instância** de uma classe
  - Objetos semelhantes pertencem a mesma classe
  - **Instância**: item de informação que é representado por um tipo

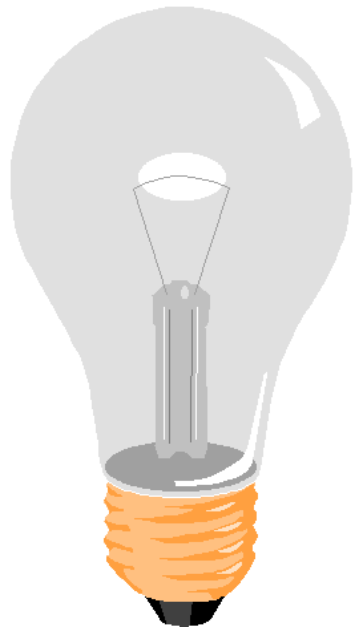
# Classes



- **Classe Lampada**
  - Atributos
    - potencia, ligada
  - métodos
    - ligar, desligar, estaLigada

<b>Lampada</b>
- ligada : boolean - potencia : double
+ ligar() : void + desligar() : void + estaLigada() : boolean

# Classes



**Nome da classe**

**Atributos**

**Métodos**



# + Programação em JAVA

```
public class Lampada{
```

```
private boolean ligada;  
private double potencia;
```

```
public void ligar(){  
    ligada = true;  
}
```

```
public void desligar(){  
    ligada = false;  
}
```

```
public boolean estaLigada(){  
    return ligada;  
}
```

```
}
```

**Atributos**

**Métodos**

# + Classes

Ponto
- x : int - y : int
+ distanciaOrigem() : double

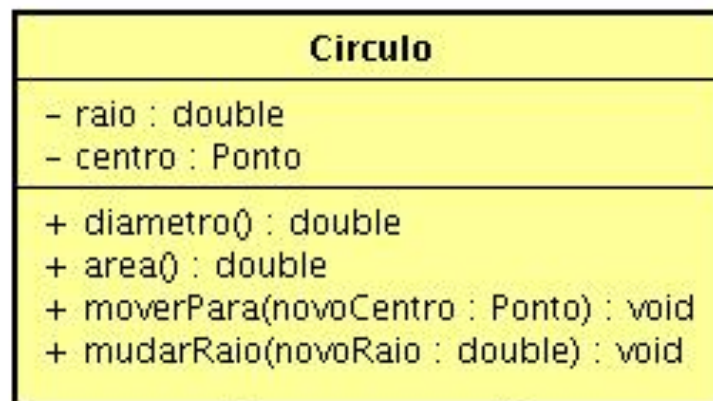
Lampada
- ligada : boolean - potencia : double
+ ligar() : void + desligar() : void + estaLigada() : boolean

Carro
- potencia : int - velocidade : double - marcha : int - cor : Cor - rotacaoMotor : int
+ ligar() : void + desligar() : void + acelerar() : void + mudarMarcha(novaMarcha : int) : void + freiar() : void + pintar(novaCor : Cor) : void

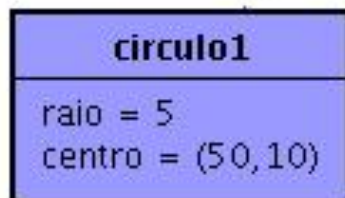
Circulo
- raio : double - centro : Ponto
+ diametro() : double + area() : double + moverPara(novoCentro : Ponto) : void + mudarRaio(novoRaio : double) : void

# + Classes e objetos

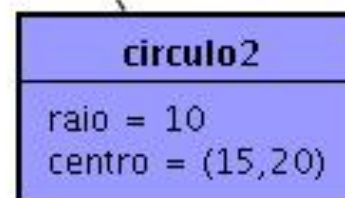
**Classe**



**Objeto**



**Objeto**



**Circulo circulo1 = new Circulo();**

**Circulo circulo2 = new Circulo();**

# + Resumo

- Objeto
  - Qualquer entidade que possui características e comportamento
- Classe
  - Descreve um tipo de objeto
  - Define atributos e métodos
- Atributo
  - Define características do objeto
- Método
  - operações que o objeto pode realizar



# + Dúvidas

# + Dúvidas

