Programação Orientada a Objetos

Prof. MsC Sílvio Bacalá Júnior

Princípios básicos de OO

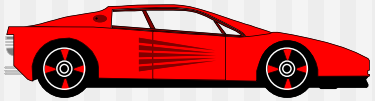
- Abstração
- Encapsulamento
- Modularidade
- Herança

Abstração

- Concentração nas características essenciais, gerenciando complexidade
- Construção de um modelo para representação de uma realidade

cliente

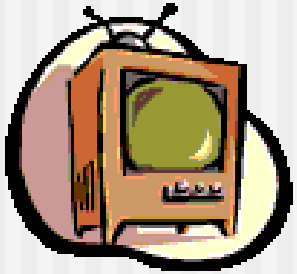
gado



automóvel

Encapsulamento

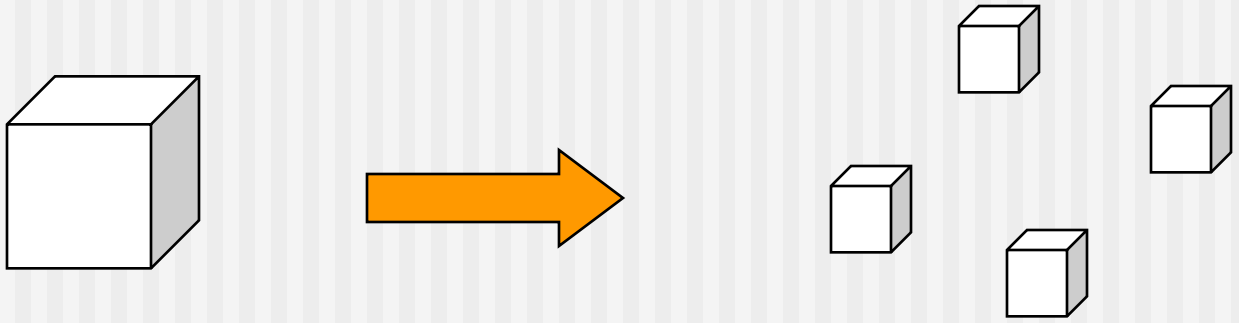
- Elimina dependência de implementação, escondendo-a do cliente
- Uso de interfaces
- Mudanças internas não têm impacto sobre os clientes



Modularidade

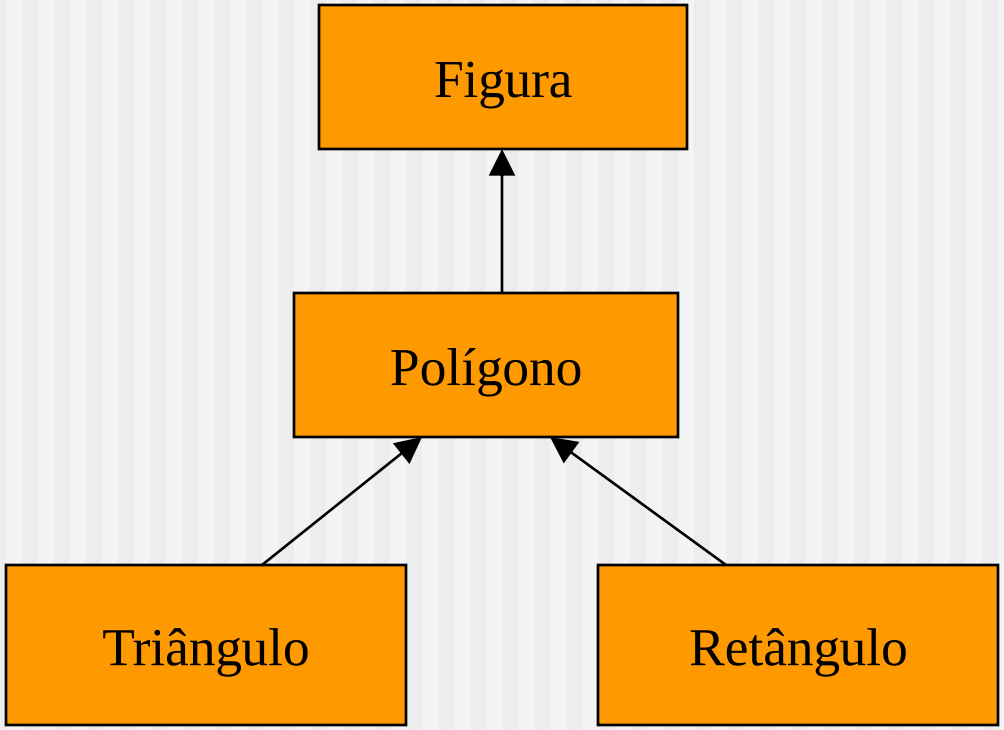
- Decomposição de um problema em pequenos pedaços, para gerenciar complexidade
- Construção de módulos desacoplados

- Dividir para conquistar



Herança

- Criação de hierarquias de abstração
- Permite ordenar hierarquias relacionadas



Domínio e Aplicação

- **Domínio**
  - composto pelas entidades, informações e processos relacionados a um determinado contexto
- **Aplicação**
  - desenvolvida para automatizar ou tornar factível as tarefas de um domínio.
  - "reflexo" de um domínio.

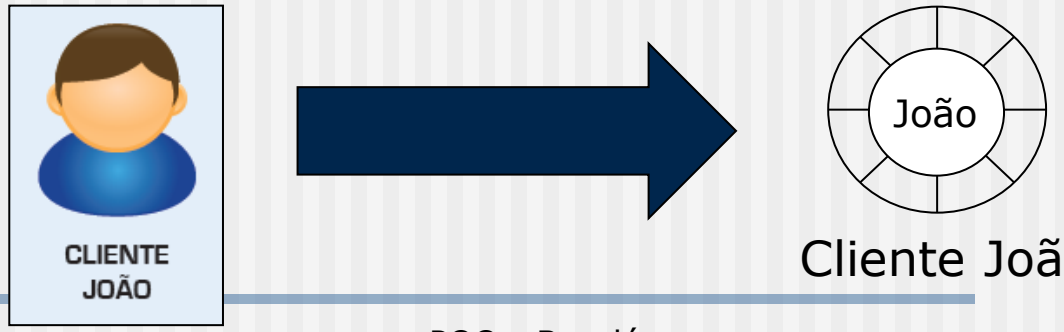
Exemplo: domínio bancário

Objetos

- As entidades identificadas no domínio devem ser representadas, numa aplicação OO por objetos.
- Uma aplicação OO é composta por objetos.
- Em geral, um objeto representa uma entidade do domínio.

Atributos

- Algumas informações do cliente como nome, data de nascimento e sexo são importantes para o banco.
- Já que são relevantes para o domínio, o objeto cliente também deve possuir essas informações.
- Esses dados são armazenados nos atributos do objeto que representa o João.



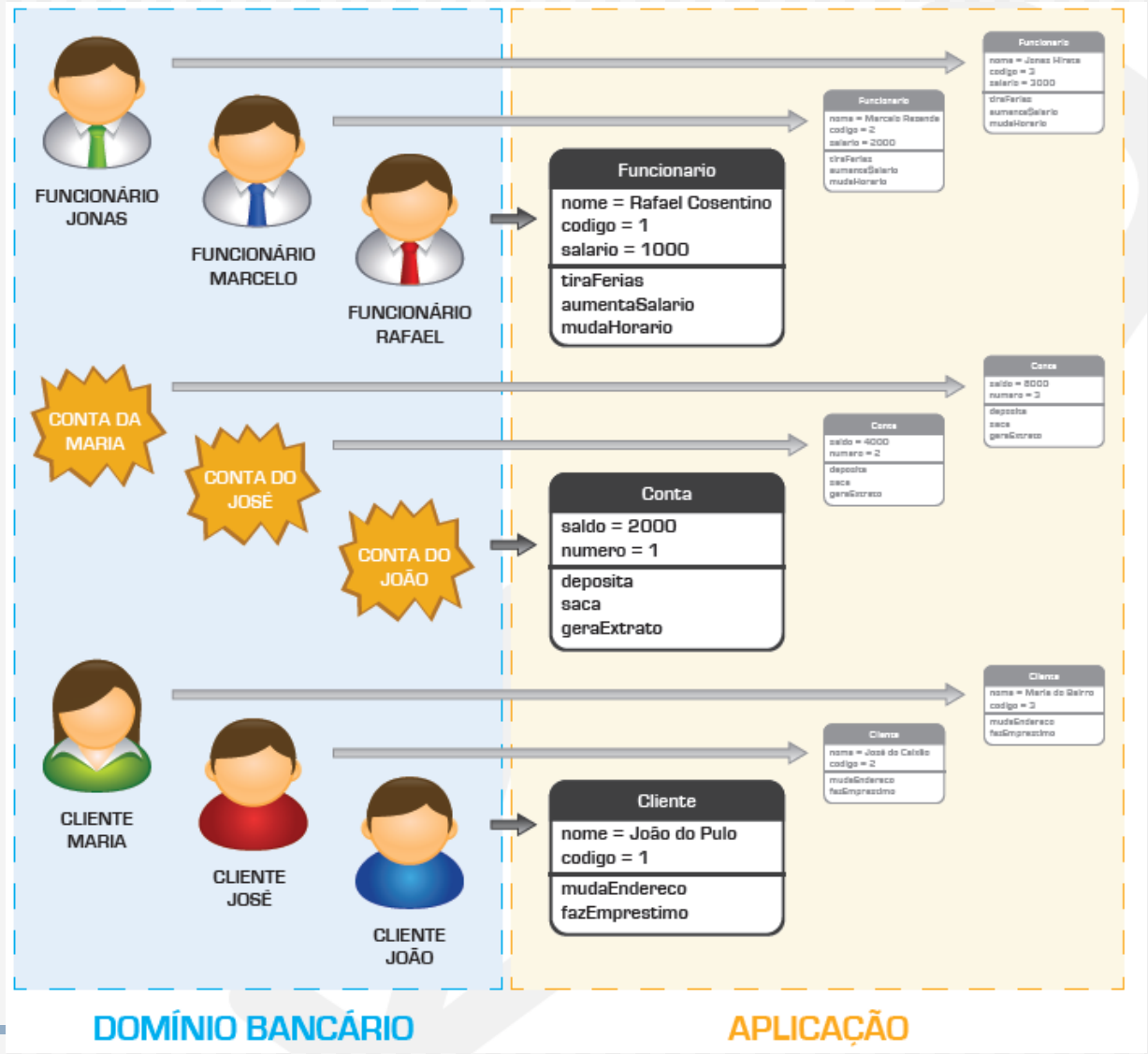
Atributos

- Variável que pertence a um objeto
- Os dados de um objeto são armazenados nos seus atributos
- O próprio objeto deve realizar operações de consulta ou alteração dos valores de seus atributos.
- Essas operações são definidas nos **métodos** do objeto.

Métodos

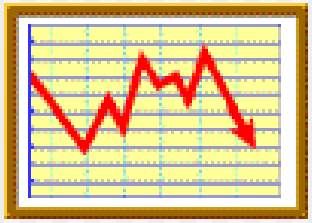
- Utilizados para possibilitar interações entre os objetos de uma aplicação.
- As tarefas que um objeto pode realizar são definidas pelos seus métodos.
- Um objeto é composto por
  - atributos e
  - métodos.

Exemplo


FUNCIONÁRIO JONAS
FUNCIONÁRIO MARCELO
FUNCIONÁRIO RAFAEL
Funcionario
nome = Rafael Cosentino
codigo = 1
salario = 1000
tiraFerias
aumentaSalario
mudaHorario
CONTA DA MARIA
CONTA DO JOSÉ
CONTA DO JOÃO
Conta
saldo = 2000
numero = 1
deposita
saca
geraExtrato
CLIENTE MARIA
CLIENTE JOSÉ
CLIENTE JOÃO
Cliente
nome = João do Pulo
codigo = 1
mudaEndereco
fazEmprestimo
DOMÍNIO BANCÁRIO
APLICAÇÃO

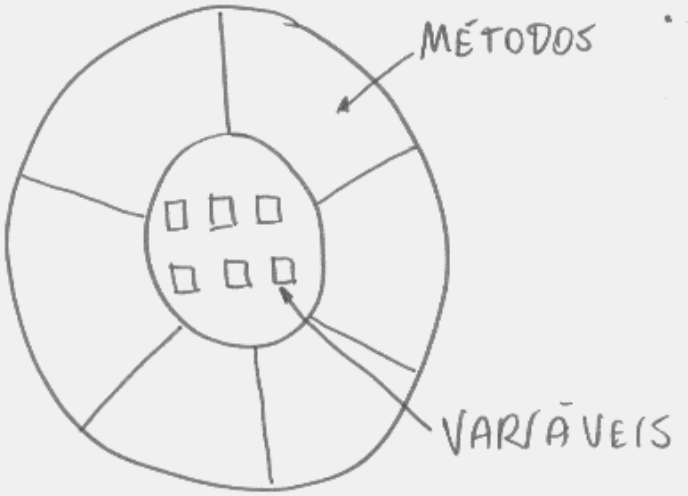
Objeto

- Modelo de um objeto real
  - entidade física, conceitual ou de software
- Possui comportamento, estado e identidade



Objetos, Métodos e Variáveis

- **OBJETO** é um pacote de software contendo dados e procedimentos (código) relacionados
- Os procedimentos são chamados **MÉTODOS**
- Os dados dos objetos são chamados **VARIÁVEIS** ou **COMPONENTES DO ESTADO**



Objeto em UML

: Cliente

Apenas o nome da classe

aquela

Apenas o nome do objeto

aquela : Cliente

Nome da classe e do objeto

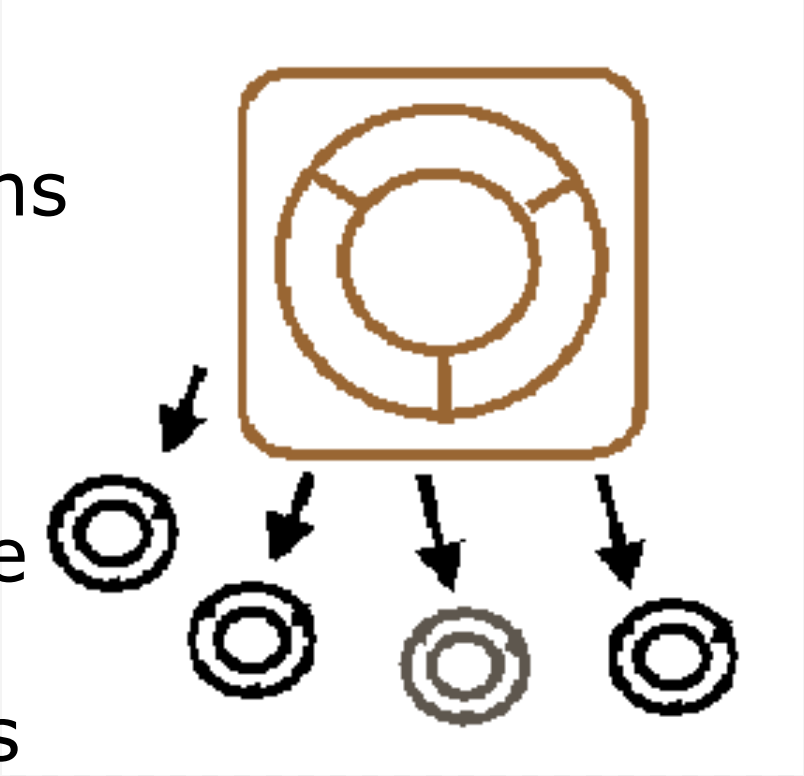
Aquela Cliente do ARIEL

Classes

- Define quais serão os atributos e métodos de um objeto.
- Definição realizada por meio de uma classe elaborada por um programador.
- A partir de uma classe, podemos construir objetos que executam a nossa aplicação.

Classe

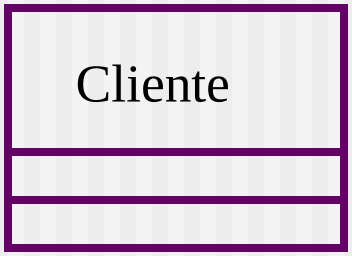
- Descrições de objetos com propriedades e comportamento comuns
- Abstração que
  - enfatiza o que é relevante
  - suprime o que não interessa
- Classes são fábricas de objetos
- Objetos são agrupados em classes



Classes e Instâncias

- **Classes** - modelos que definem os métodos e as variáveis a serem incluídas em um tipo particular de objeto.
- Objetos que pertencem a uma classe são chamados de **INSTÂNCIAS** desta classe e contêm valores particulares para as suas variáveis.
- Variáveis de um objeto são chamadas de **VARIÁVEIS DE INSTÂNCIA**

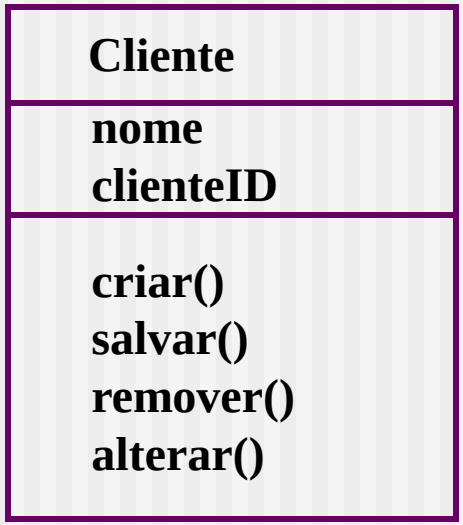
Classe em UML



Nome da Classe →

Atributos →

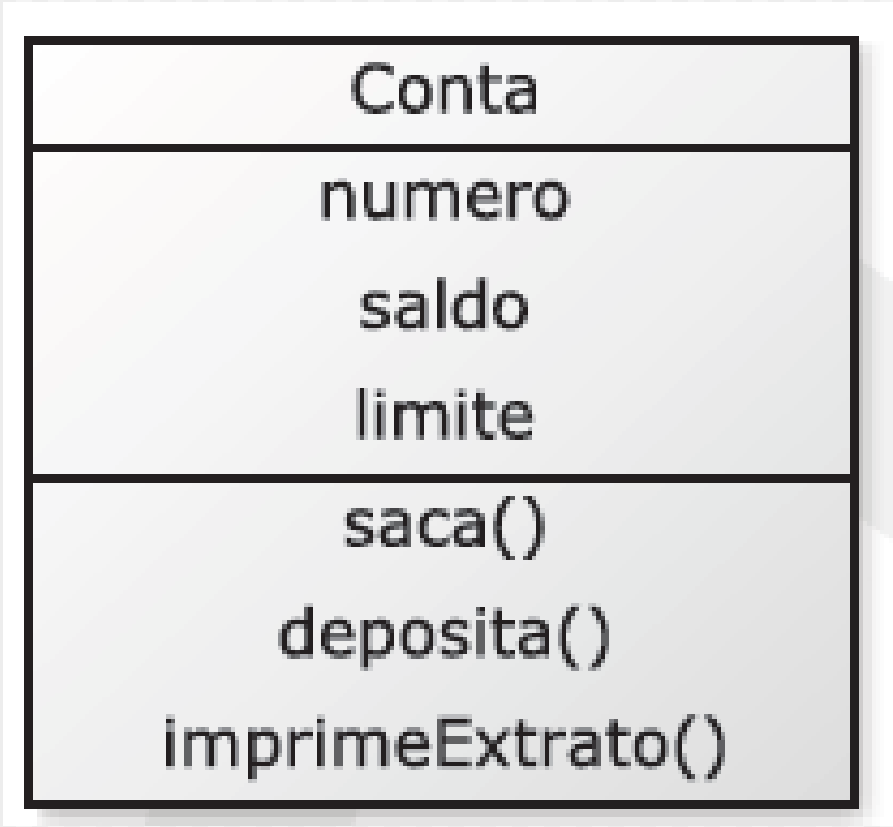
Operações →



estrutura

comportamento

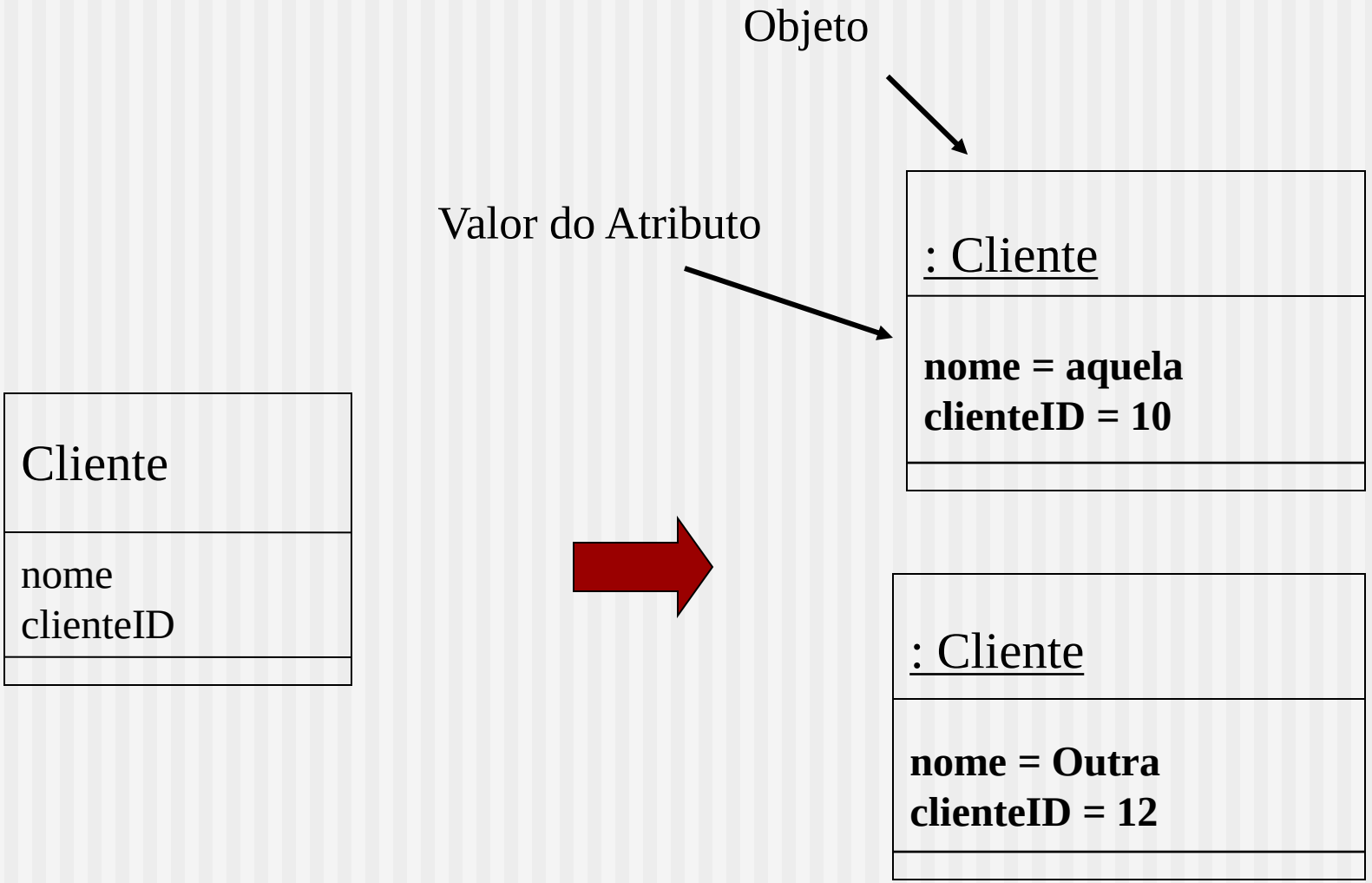
Representação de Classes



Atributo

- Armazenam estado dos objetos
  - são coisas que um objeto “sabe”
- Propriedades de um objeto
  - Variáveis de instância

Atributo em UML



Operação

- Modela comportamento das classes
- São coisas que uma classe "faz"
- Serviços que os objetos oferecem a outros objetos

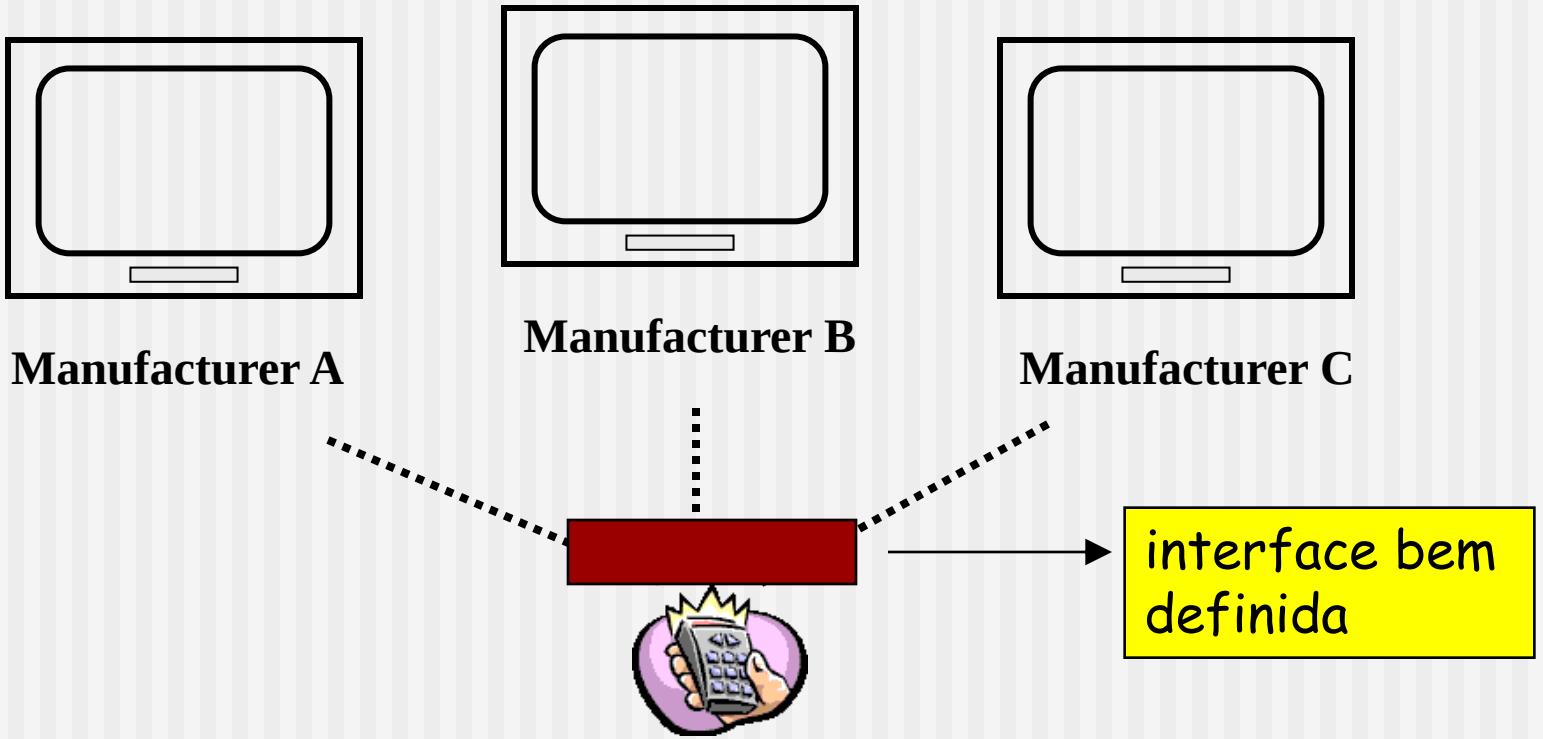
Classes e Objetos em Java

```
class Aviao {
    ..
    Asa asa[2];
    Propulsor propulsor[];
    LemeDeDirecao leme;
    ..
    void pouse(..) {
        ..
        leme.reto();
        ..
        asa[1].flapPBaixo(..);
        asa[2].flapPBaixo(..);
        ..
        propulsor.desligue(..);
        ..
    }
}
```

```
class Asa {
        ..
        void flapPBaixo(Grau graus) {
            ..
        }
        void flapPCima(Grau graus) {
            ..
        }
}
class Propulsor {
        Int potencia;
        void desligue(..) {
            ..
        }
        void ligue(..) {
            ..
        }
}
```

Polimorfismo

- Escondendo diferentes implementações através de uma única interface



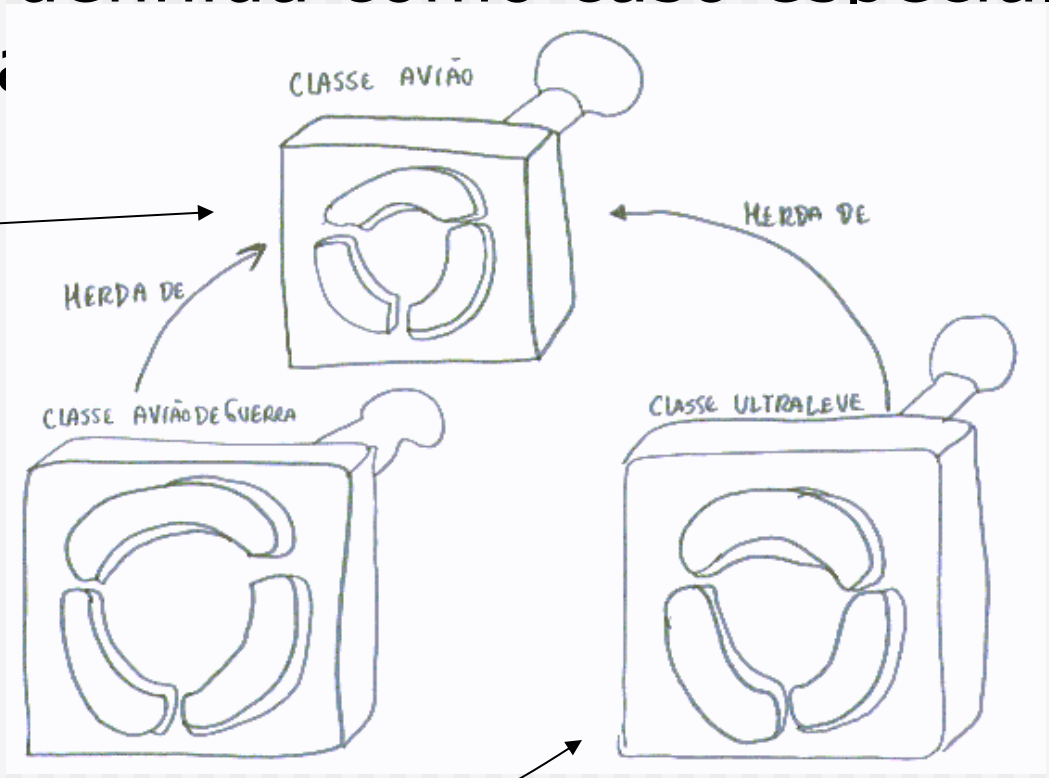
Fonte: Rational

Herança

- Mecanismo através do qual uma classe de objetos pode ser definida como caso especial de uma classe ma

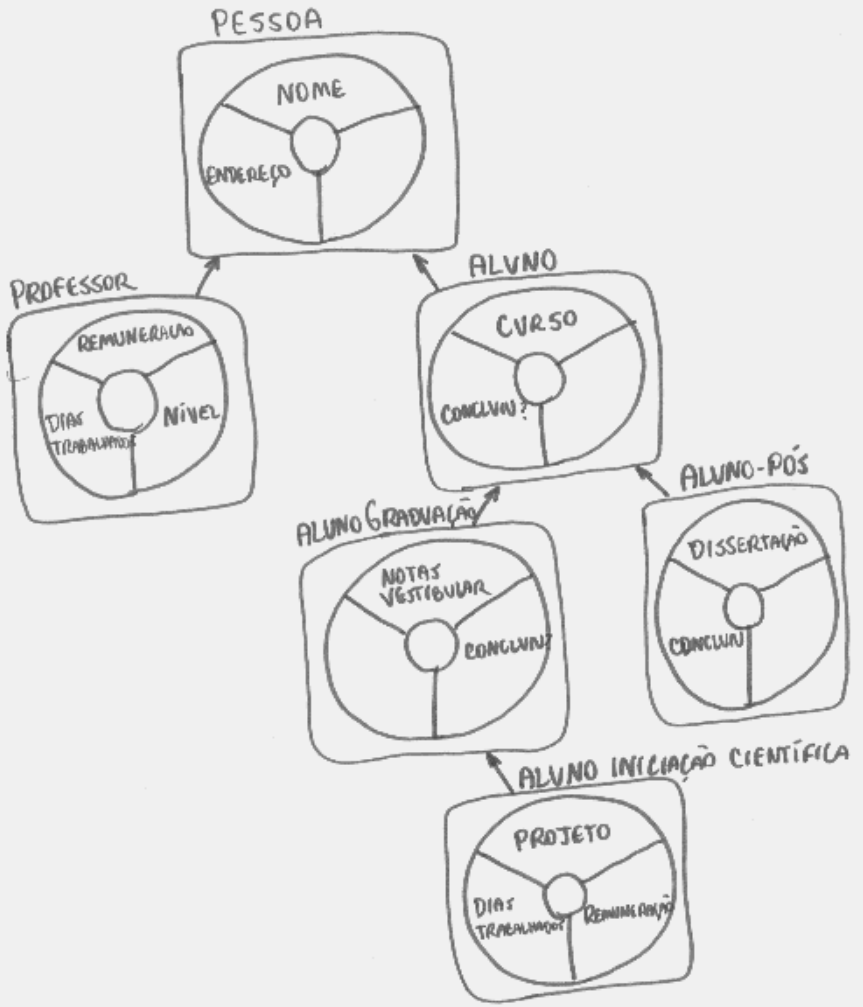
A classe mais geral é chamada de **SUPERCLASSE**

Casos especiais de uma classe são chamados de **SUBCLASSE**.



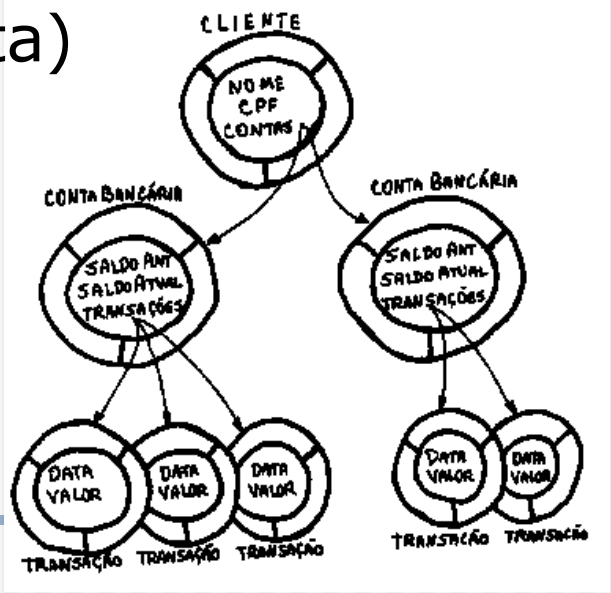
Herança e Polimorfismo

- Permitem que objetos sejam tratados de forma homogênea e segura um conjunto consistente de abstrações



Exercício

- Represente os objetos que são necessários para fazer uma aplicação OO para controlar um:
  1. Bar ou Restaurante
  2. Escola de Inglês ou Natação
  3. Pequeno Armazém que trabalha com contas de clientes (tipo caderneta)

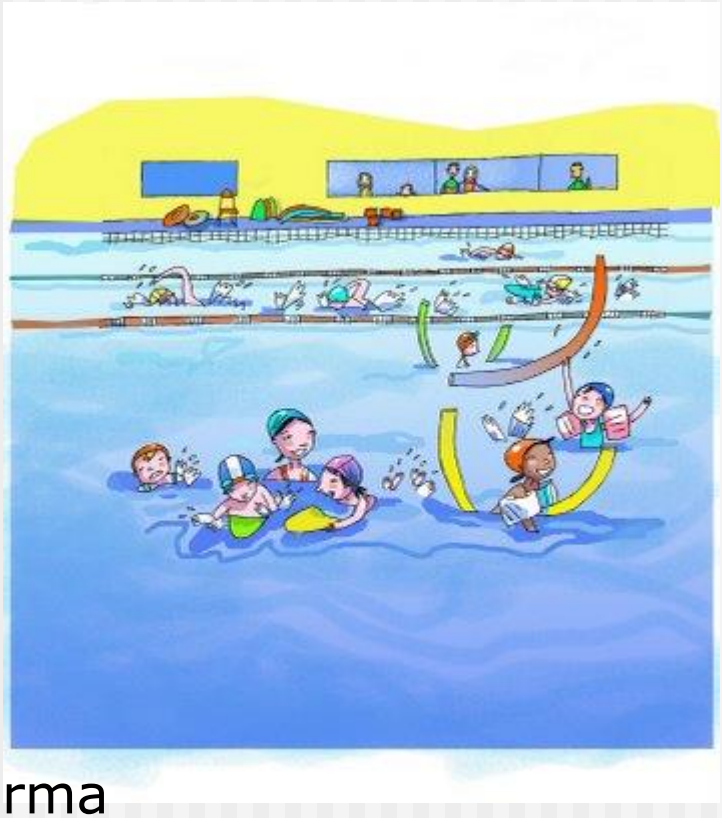


Bar ou Restaurante

- Conceitos do Domínio
  - Produtos que compõem o cardápio
  - Contas associadas às mesas
  - Consumo de produtos associados às contas das mesas
  - Garçons responsáveis pelas contas
  - Etc...
- Operações:
  - Abrir conta da mesa
  - Registrar pedidos na conta
  - Fechar conta
  - Receber conta
  - Etc...

Escola de Inglês ou Natação

- Conceitos do Domínio
  - Alunos
  - Professores
  - Turmas
  - Pagamentos
  - Etc...
- Operações:
  - Matricular alunos em turma
  - Receber pagamento de aluno
  - Escalar professor para uma turma
  - Controlar frequência de alunos
  - Etc...



Armazém

- Conceitos do Domínio
  - Clientes
  - Caderneta do Cliente
  - Produtos
  - Estoque
  - Etc...
- Operações:
  - Registra venda na caderneta
  - Receber pagamento do cliente
  - Consultar estoque de produtos
  - Etc...

Benefícios do Paradigma de Orientação a Objetos

- Favorece modularidade, manutenibilidade e reuso
- Aproxima-se do mundo real
- Uso do mesmo conceito em todas as fases do desenvolvimento