

The background features a series of concentric circles in light gray, some solid and some dashed, creating a subtle pattern. Overlaid on this is a large, solid red speech bubble with a pointed bottom, which serves as a container for the title text.

Introdução à Linguagem Java

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large, solid red speech bubble is centered on the page, pointing downwards. The text is white and centered within the bubble.

Herança e Polimorfismo



POO

- Baseado em 4 conceitos:
 1. Abstração
 2. Encapsulamento
 3. Herança
 4. Polimorfismo

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large, solid red speech bubble is centered on the page, pointing downwards.

Herança

Herança



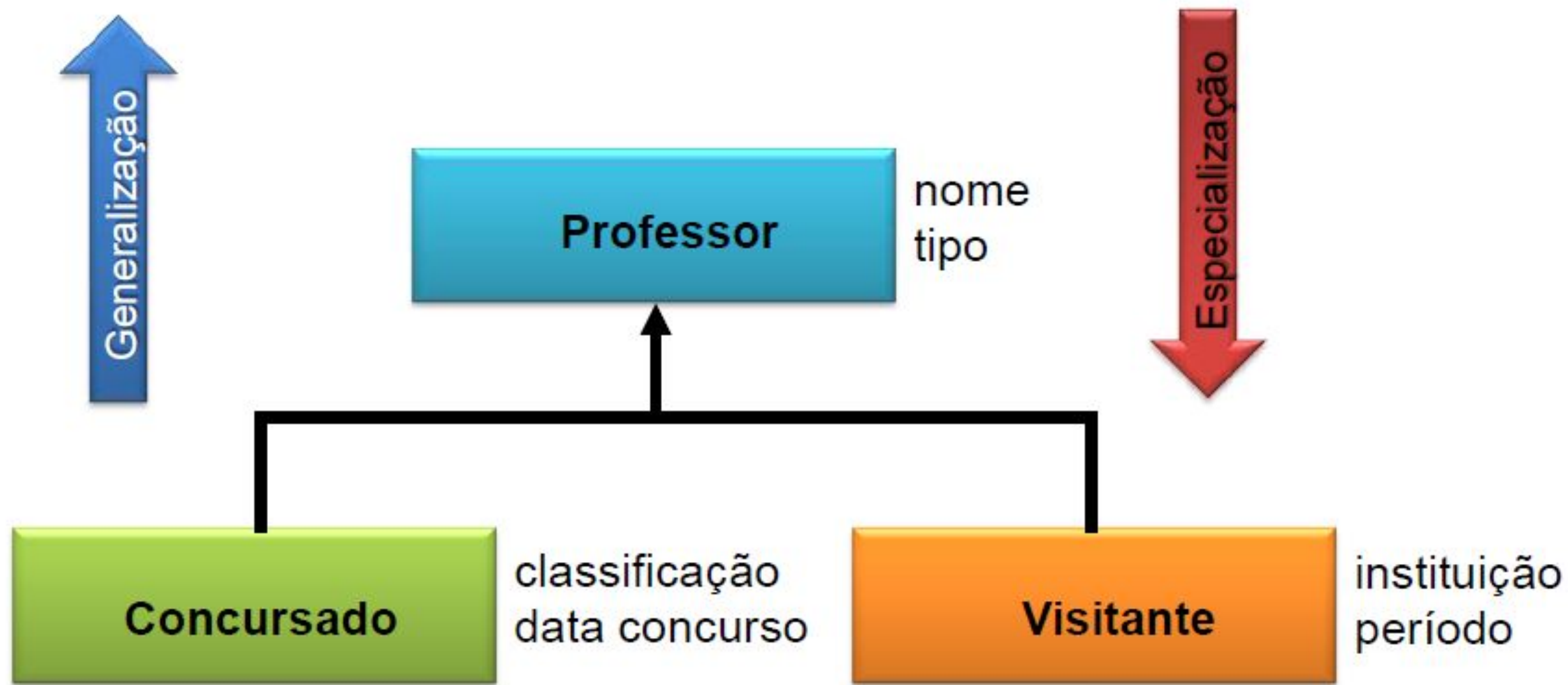
Fonte: https://istoe.com.br/189303_COMO+ADMINISTRAR+A+HERANCA/

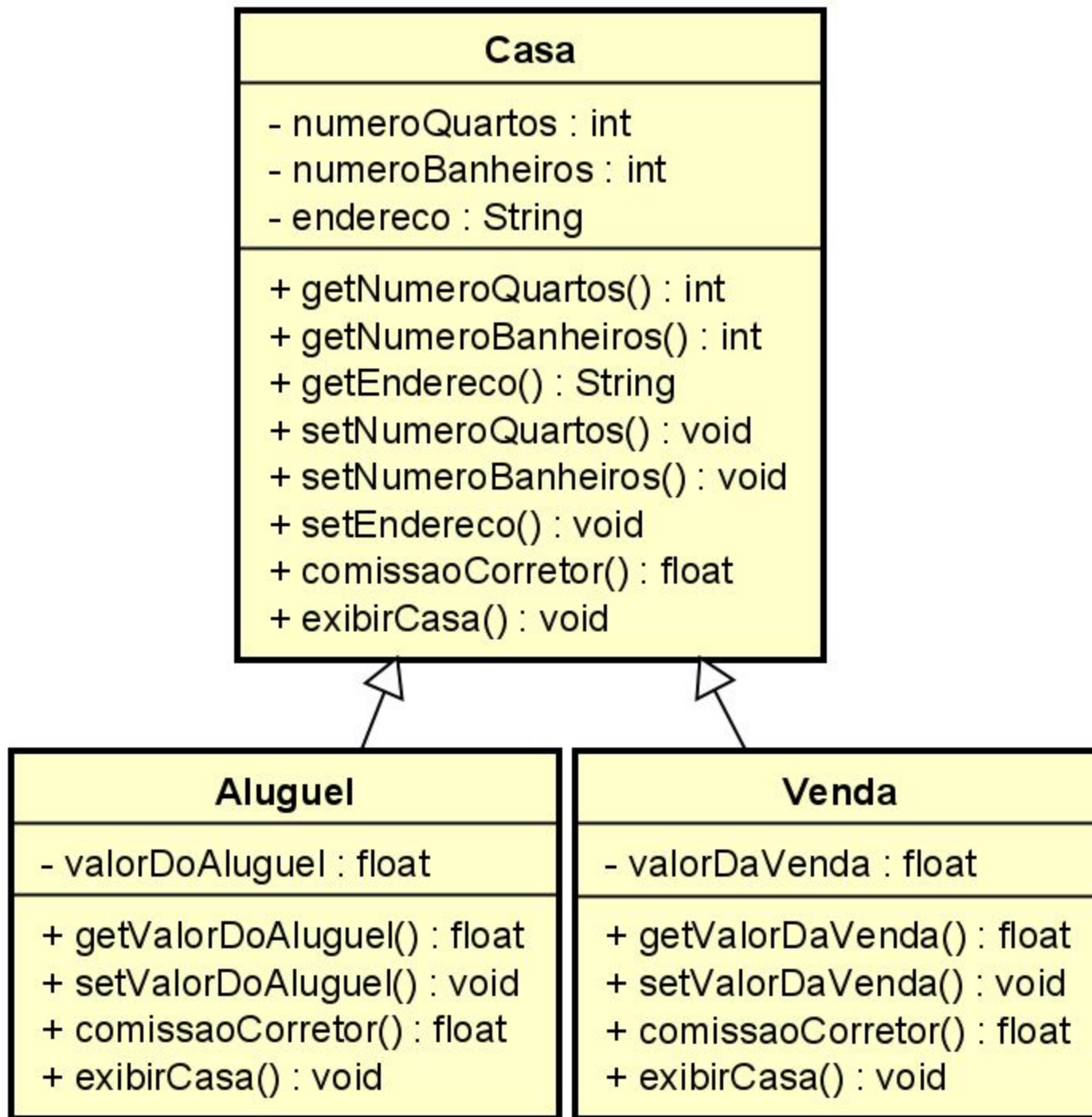
Herança

- Herança é uma relação hierárquica entre classes permitindo aproveitar e compartilhar o que elas tem em comum.
- Vantagens:
 1. Reutilização de código/Facilidade de manutenção
 2. Permite maior semelhança com mundo real

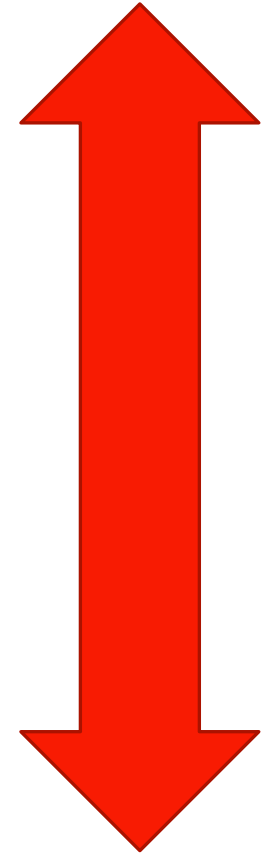
Herança

- Em uma herança:
 - A classe herdada é chamada de SUPERCLASSE
 - A classe que herda é chamada de SUBCLASSE
- Uma SUPERCLASSE é mais geral que uma SUBCLASSE
 - Abrange mais tipos de objetos
- Uma SUBCLASSE é mais específica
 - Representa um grupo menor e mais especializado





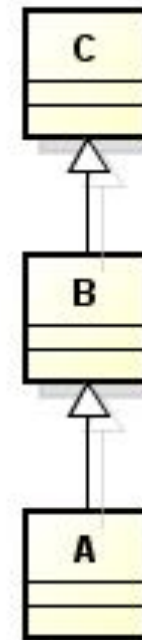
Generalização



Especialização

Herança

- *Se uma classe A herda de B e uma classe B herda de C, então A também herda de C*



Herança

- Para se explicitar o uso da HERANÇA em java, usasse o operador EXTENDS

```
public class Aluguel extends Casa
```

Herança

- Atributos e métodos **privados** são herdados, mas não podem ser acessados diretamente. Deve-se usar os métodos *get* (acessador) e *set* (alterador);
- Uma instância de uma subclasse é uma instância de todas as suas superclasses;
- Objetos da subclasse comportam-se como objetos da superclasse;
- Objetos da subclasse podem ser usados no lugar de objetos da superclasse

Herança

- Em JAVA, só é permitido herança simples, ou seja:
NENHUMA CLASSE PODE TER MAIS DE UM PAI no mesmo nível!
- Existem linguagens que suportam herança múltipla como C++...

Herança

- *Vantagens...*

- Reuso

- Atributos e funções são passados a diante

- Manutenção

- Em caso de alteração em método, basta alterar na classe mais geral

Herança

- Vimos que o operador **THIS** faz referência à elementos (atributos, método, e ao próprio objeto) da classe em questão...
- Ex:
 - `this.nome = nome;`
 - `this.Metodo1();`
- Para referenciar elementos da classe-pai, usa-se o operador **SUPER**

Exercício

- Implemente as classes Aluguel e Venda

Sobrecarga e Sobescrita de Métodos

Sobrecarga de métodos

- Métodos que tem o mesmo nome mas diferentes argumentos.

- Exemplo:

```
public int soma(int n1, int n2){  
    return n1+n2; }
```

```
public float soma(float n1, float n2){  
    return n1+n2; }
```

- Regra básica para sobrecarga:
 - métodos sobrecarregados não podem ter a mesma assinatura.

Sobrecarga de Método

```
public int metodo( int a, double b );
```

```
public void metodo( int a, long b );  
// OK - tipos dos parâmetros diferentes
```

```
public void metodo( int a, long b, float c );  
// OK - número de parâmetros diferente
```

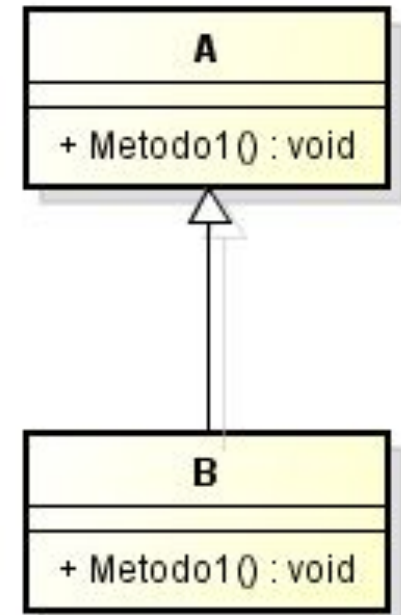
```
public void metodo( double a, int b );  
// OK - ordem de parâmetros diferente
```

```
public long metodo( int x, double b );  
// Erro - é igual a 1ª definição
```

Sobrescrita de Método

- Dada uma hierarquia....
Permite **REESCREVER** em uma classe filha, um determinado método existente na classe pai!

- Em uma superclasse A eu tenho:
 - `public void Metodo1();`
- Em uma subclasse B eu tenho:
 - `public void Metodo1();`



Comparação

- Sobrecarga:

1. Assinaturas diferentes
2. Mesma classe
3. Exemplo:
`impExtrato(int mes)`
`impExtrato(int ini, int fim)`

- Sobrescrita:

1. Assinaturas iguais
2. Classes diferentes
3. Subclasse tem nova implementação do método herdado

Exercício

- Implemente o método `comissaoCorretor` na classe `Aluguel e Venda`.
- Leve em consideração que o valor da comissão em caso de aluguel é igual à um mês da mensalidade. Já nas vendas, a comissão corresponde à 6% do valor do imóvel.

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large red speech bubble is centered on the page, with the word "Polimorfismo" written inside in white.

Polimorfismo

Polimorfismo

- Baseado nos conceitos de sobrecarga e sobrescrita de métodos, podemos definir um conceito importante em O.O chamado de **POLIMORFISMO**
- Poli = Vários ; morfismo = formas
- Mesmo método se comportando de forma diferente = polimorfismo. (Vinculação dinâmica de métodos!)

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. In the center, there is a large red speech bubble with a white border. Inside the bubble, the text is written in white.

Por hoje é só!

Não esqueçam de fazer os exercícios!

Bons estudos!