UNIP
UNIVERSIDADE PAULISTA

LINGUAGEM DE PROGRAMAÇÃO ORIENTADA À OBJETOS – LPOO

Aula 10 – Herança e Interfaces

HERANÇA

A herança é uma forma eficaz de reutilização de código

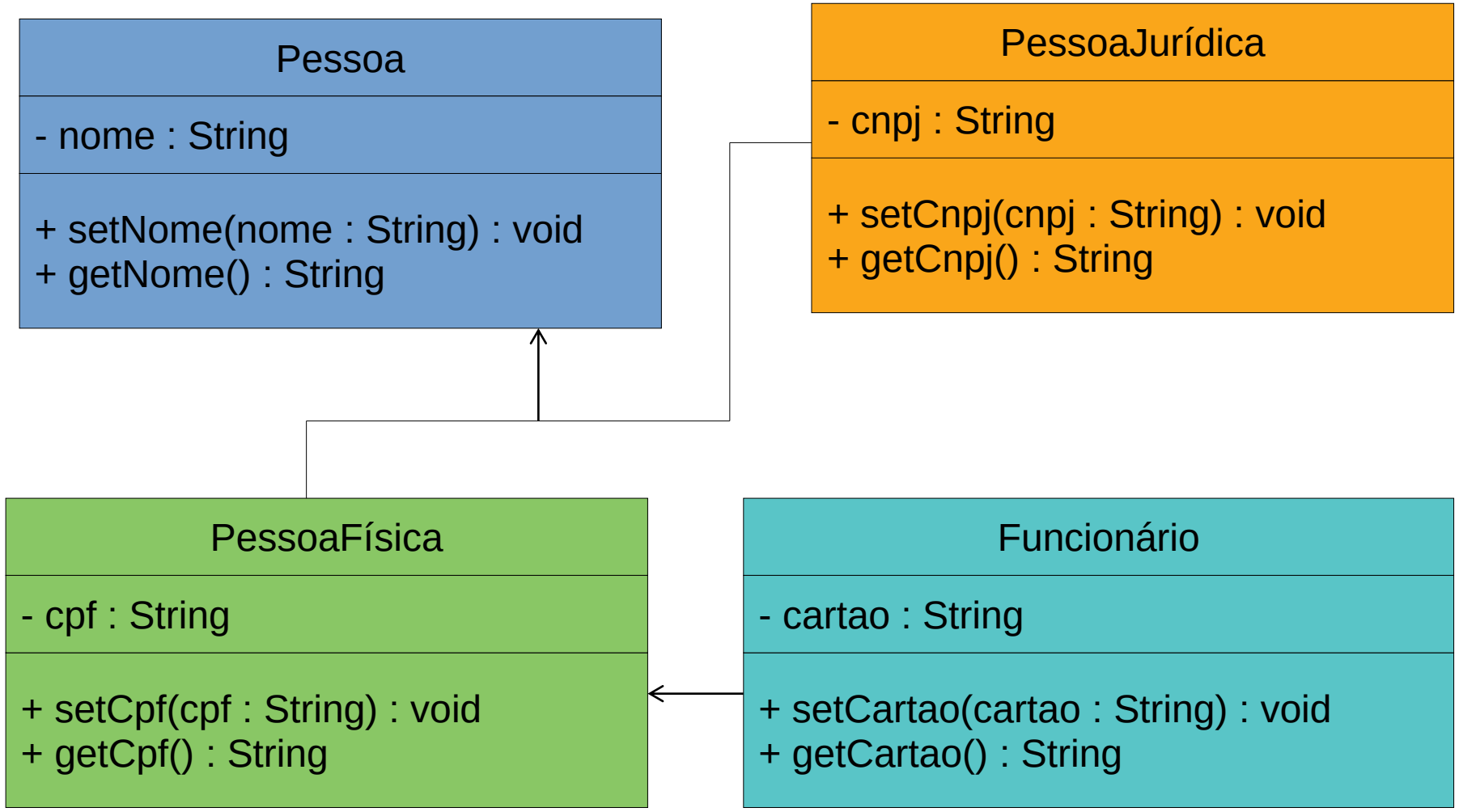
Ocorre quando uma classe passa a **herdar características** (**variáveis e métodos**) definidas em outra classe.

A classe genérica é denominada **superclasse, classe base ou classe mãe**.

As classes específicas são denominadas **subclasses, classes derivadas ou classes filhas**.

HERANÇA

- **Exemplo:**

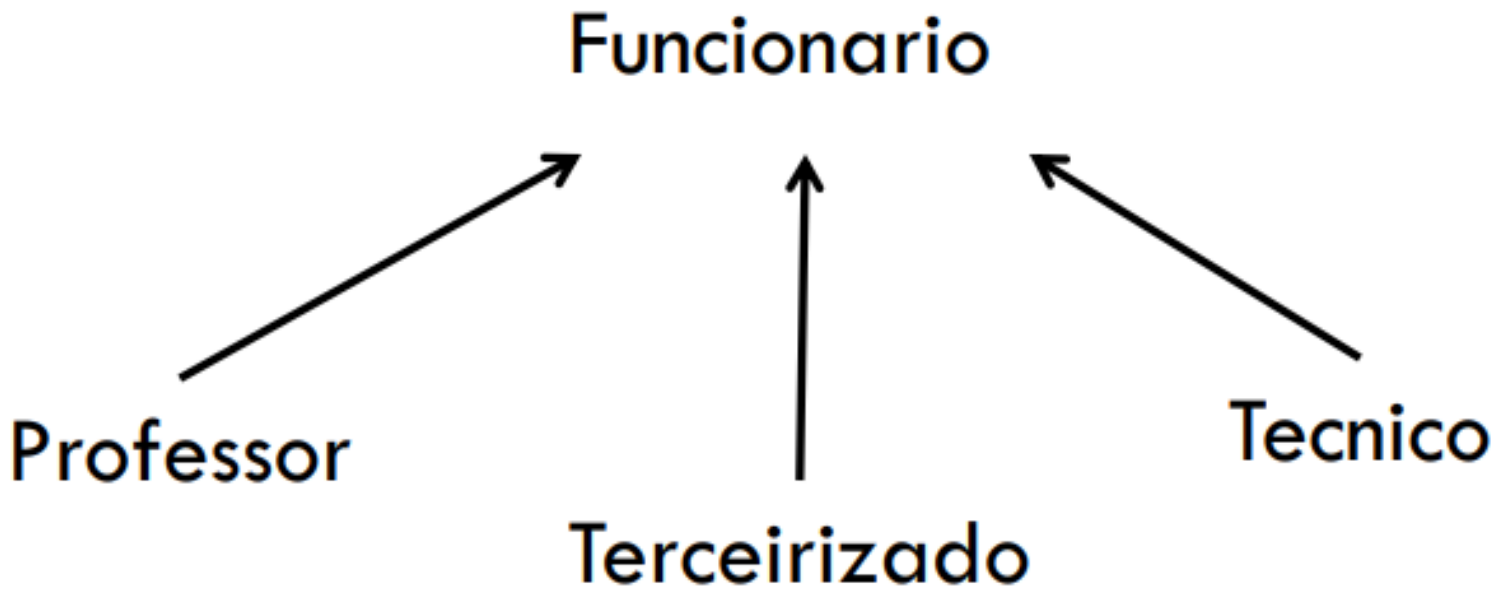


HERANÇA

- **Exemplo**: Sistema de RH de uma empresa.

  - Temos a classe Funcionário e outras classes para cada atividade específica

  (Professor, terceirizado, técnico...)

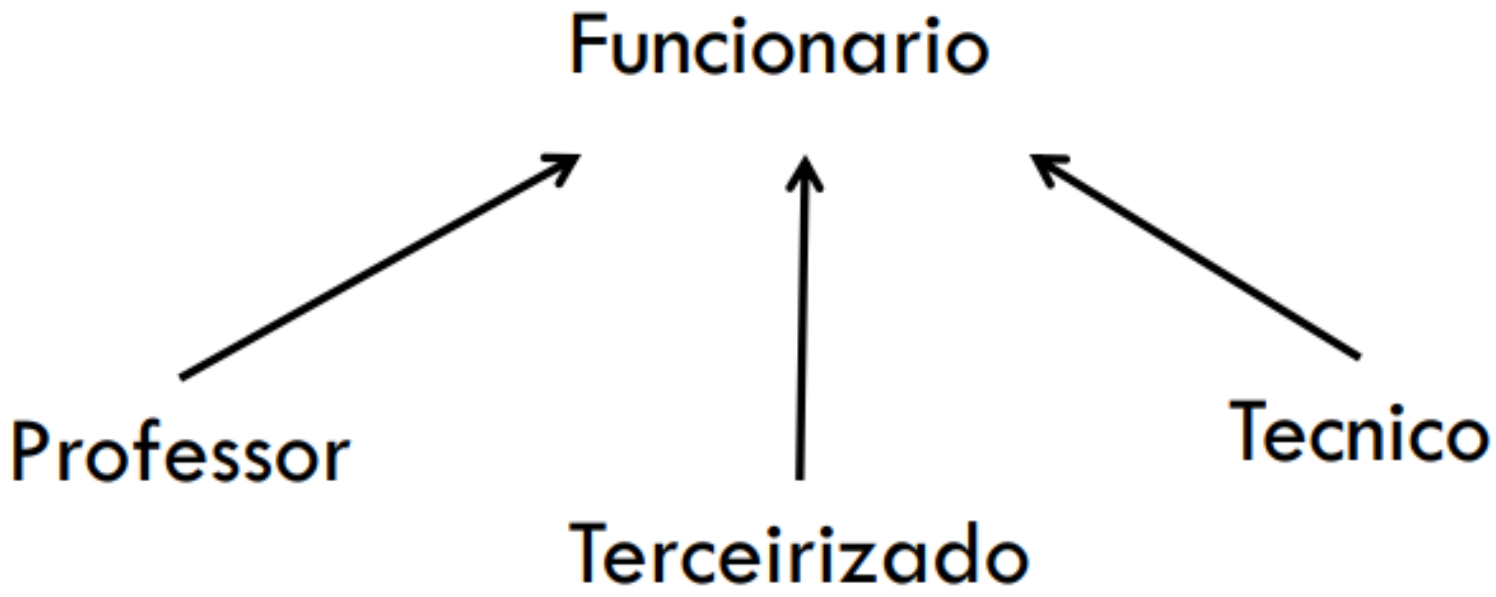


HERANÇA

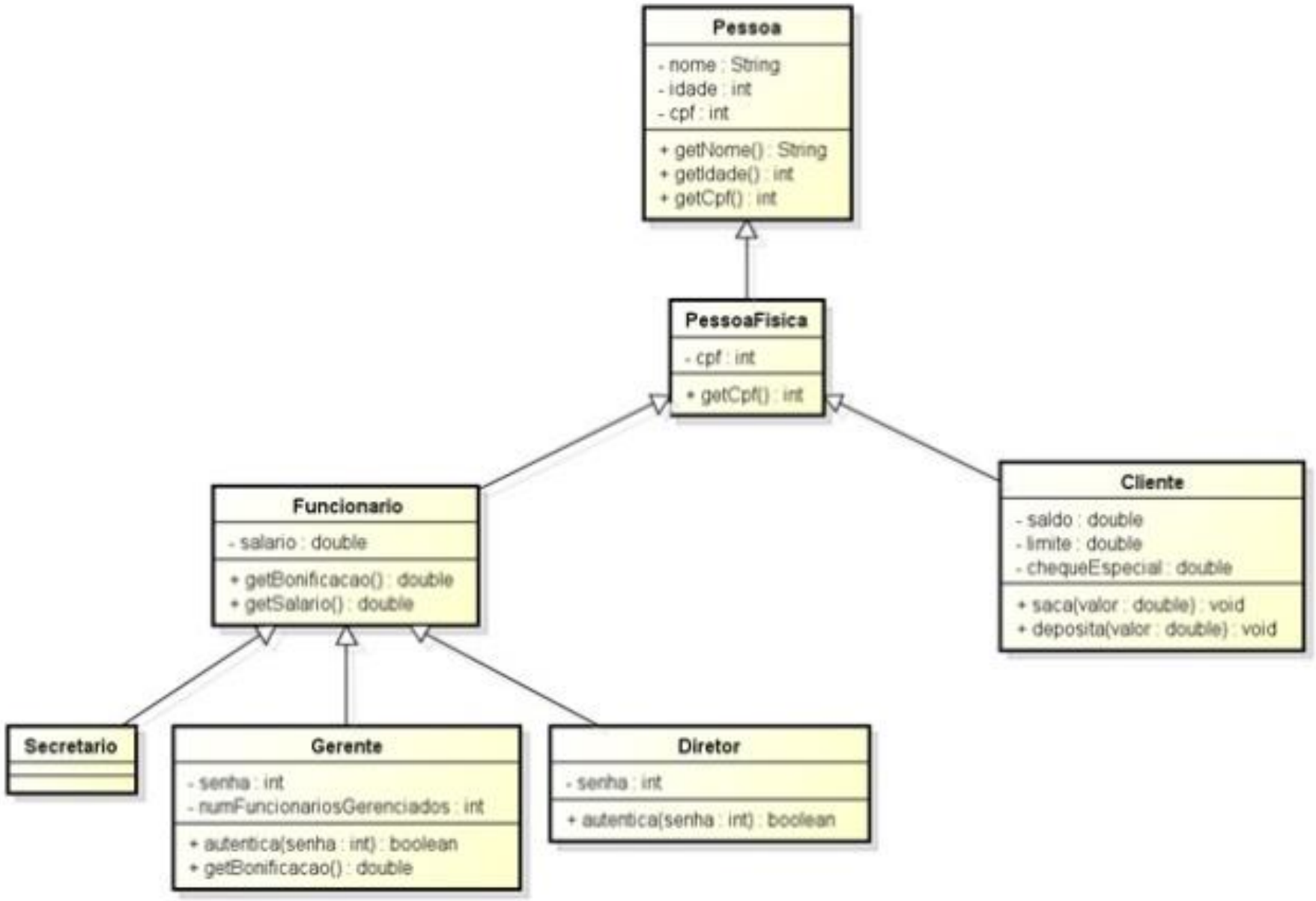
- **Exemplo**: Sistema de RH de uma empresa.

  - Temos a classe Funcionário e outras classes para cada atividade específica

  (Professor, terceirizado, técnico...)



HERANÇA


Pessoa
- nome : String
- idade : int
- cpf : int
+ getNome() : String
+ getIdade() : int
+ getCpf() : int
PessoaFisica
- cpf : int
+ getCpf() : int
Funcionario
- salario : double
+ getBonificacao() : double
+ getSalario() : double
Cliente
- saldo : double
- limite : double
- chequeEspecial : double
+ saca(valor : double) : void
+ deposita(valor : double) : void
Secretario
Gerente
- senha : int
- numFuncionariosGerenciados : int
+ autentica(senha : int) : boolean
+ getBonificacao() : double
Diretor
- senha : int
+ autentica(senha : int) : boolean

HERANÇA

Com a criação dessa hierarquia, surgem dois novos conceitos:

- Superclasse (classe mãe)

- Subclasse (classe filha)

O relacionamento criado entre uma superclasse e uma subclasse pode ser chamado de relacionamento **“é um”.**

HERANÇA

- Dizemos que a subclasse estende da superclasse.
- Então, para criarmos uma herança em nosso sistema, utilizamos a palavra reservada **extends**.
  - ***Ex:*** *public class Professor* ***extends*** *Funcionario { ... }*
- No exemplo acima, dizemos que o Professor **"é um"** Funcionario. Temos como superclasse o Funcionario e subclasse o Professor.

INTERFACES

- Funciona de maneira similar a classes, entretanto:

- Não permite implementação de nenhum método.

- **Contem apenas as especificações dos métodos.**

INTERFACES

Permite estabelecer um "**Contrato**" entre as classes: **que define tudo o que uma classe deve fazer.**

Um padrão é definido através de especificações ou contratos.

Nas aplicações orientadas a objetos, podemos criar um "contrato" para definir um determinado conjunto de métodos que deve ser implementado pelas classes que "assinarem" este contrato.

INTERFACES

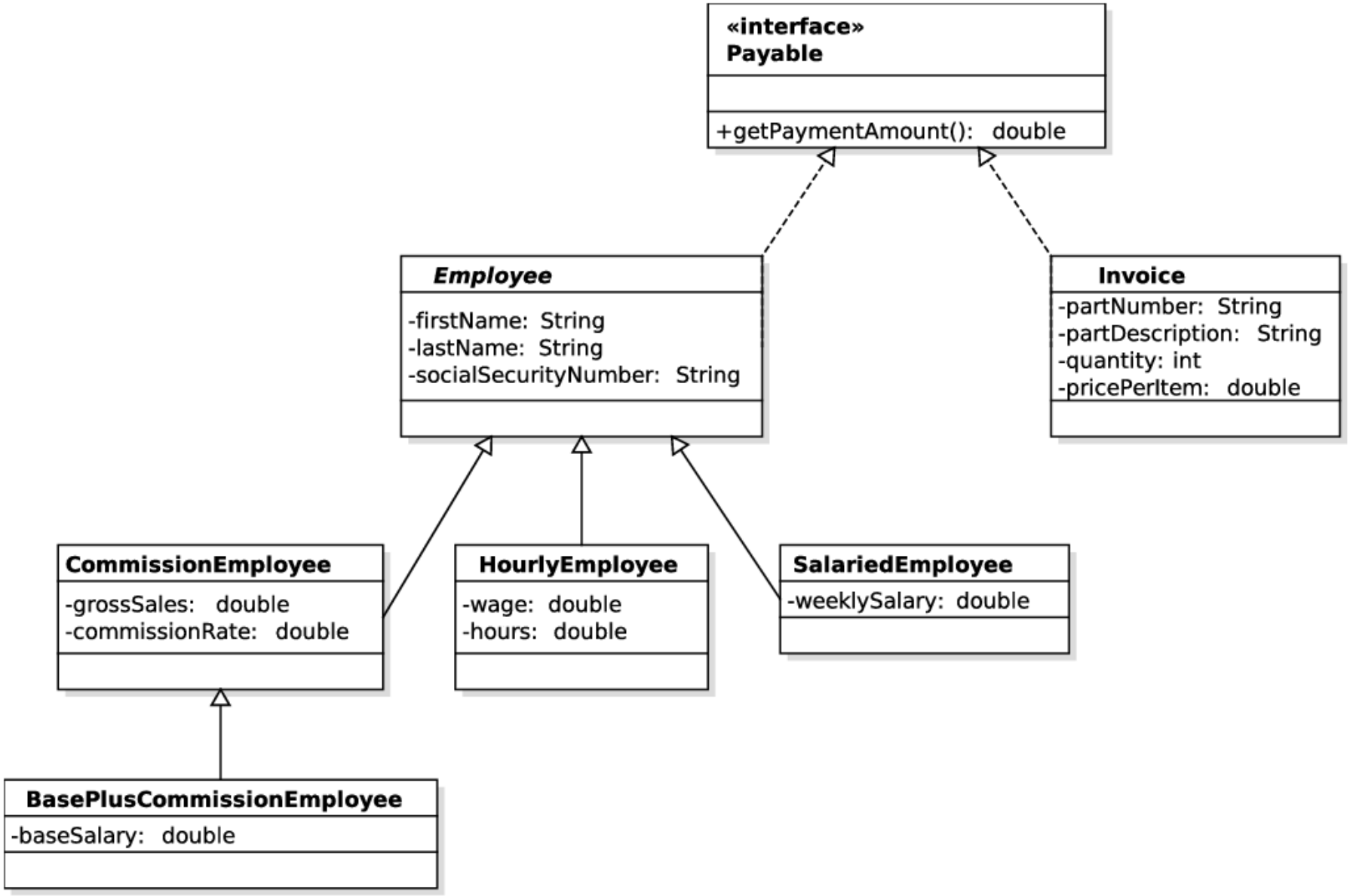
- Em orientação a objetos, um contrato é chamado de **interface**.
- Uma interface pode definir uma série de métodos, mas nunca conter implementação deles.
- Ela só expõe **o que o objeto deve fazer**, e não **como ele faz**, nem o que ele tem.
- **Como ele faz** vai ser definido em uma **implementação** dessa interface.

INTERFACES

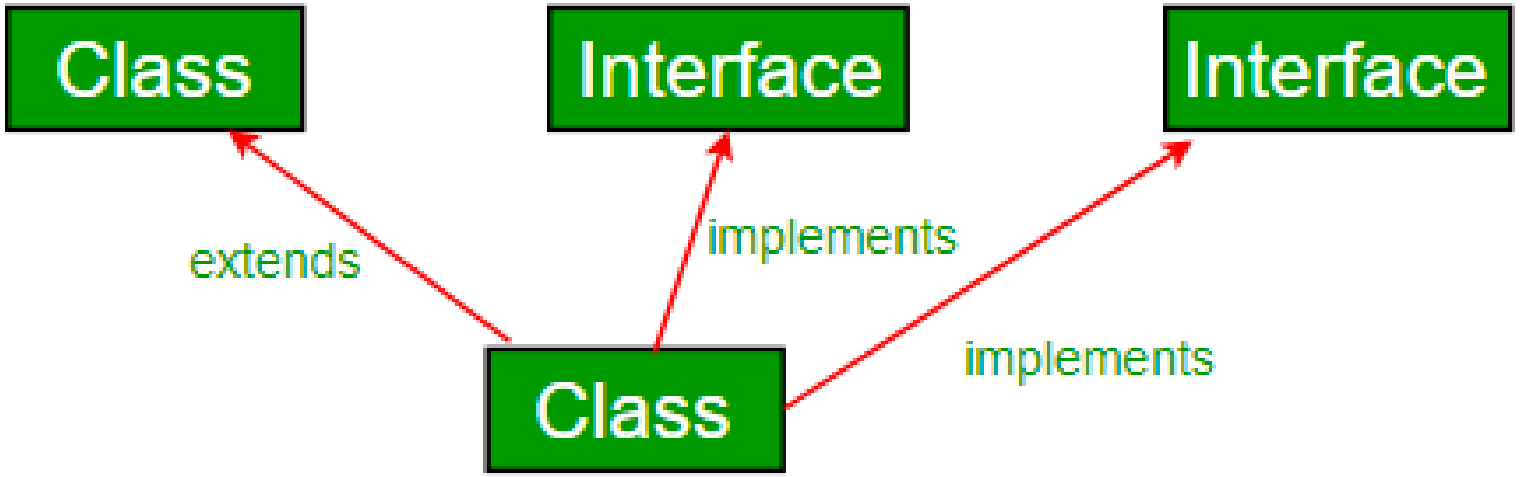
Implementação:

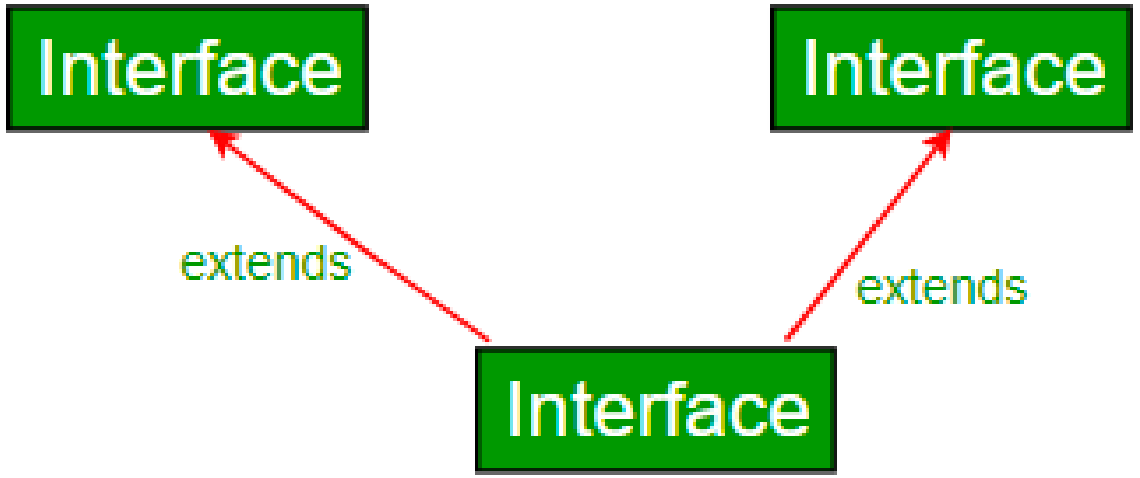
```
public interface ClienteDAO {

// especificação (assinatura) dos metódos

}
```

INTERFACES


«interface»
Payable
+getPaymentAmount(): double
Employee
-firstName: String
-lastName: String
-socialSecurityNumber: String
Invoice
-partNumber: String
-partDescription: String
-quantity: int
-pricePerItem: double
CommissionEmployee
-grossSales: double
-commissionRate: double
HourlyEmployee
-wage: double
-hours: double
SalariedEmployee
-weeklySalary: double
BasePlusCommissionEmployee
-baseSalary: double

HERANÇA x INTERFACES





EXERCÍCIOS

1) Implemente a classe Funcionario com nome, salario e os métodos **addAumento(double valor), ganhoAnual() e exibeDados() -** imprime os valores do funcionário.

a) crie a classe Assistente, que também é um funcionário, e que possui um número de matrícula (faça os métodos GET e SET). Sobrescreva o método exibeDados().

b) sabendo que os Assistentes Técnicos possuem um bônus salarial e que os Assistentes Administrativos possuem um turno (dia ou noite) e um adicional noturno, crie as classes Tecnico e Administrativo e sobrescreva o método ganhoAnual() de ambas as classes (Administrativo e Tecnico)