

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São Carlos

Análise e Desenvolvimento de Sistemas

Programação Orientada a Objetos (POO3)

Interfaces

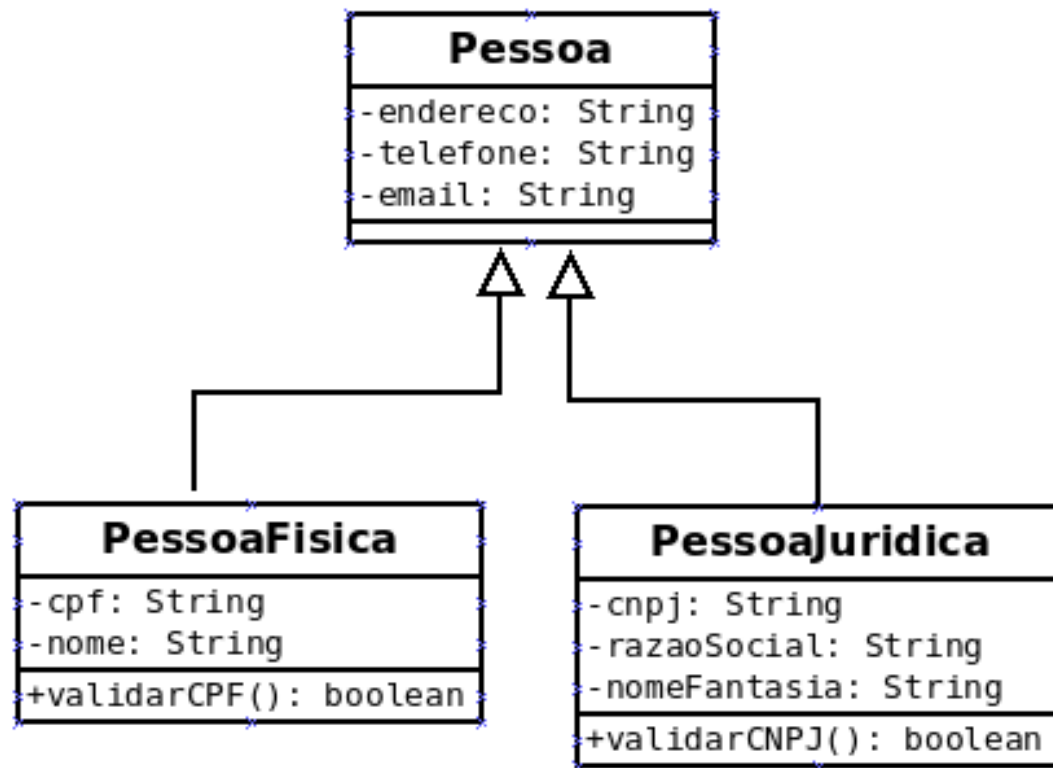


Pablo Dalbem

dalbem@ifsp.edu.br

Recapitulando...

Herança: implementa a relação “*é um*”



Recapitulando...

Classe abstrata: não pode ser instanciada.

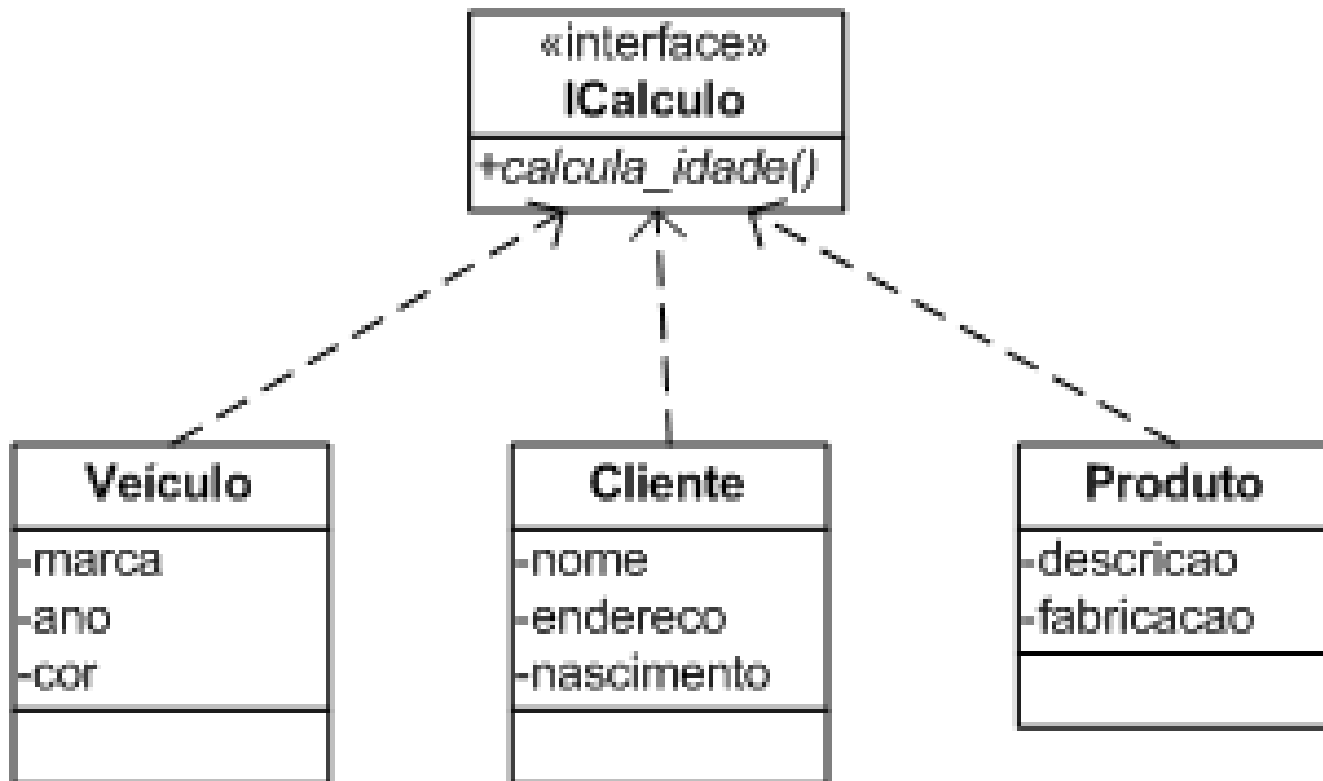
Serve como ***modelo***, ***contrato*** para as classes filhas.

Garante que subclasses possuam comportamentos mandatórios.

```
public abstract class Pessoa {  
    private String endereco;  
    private String telefone;  
    private String email;  
  
    public abstract void mostrarDados();  
  
    ....  
}
```

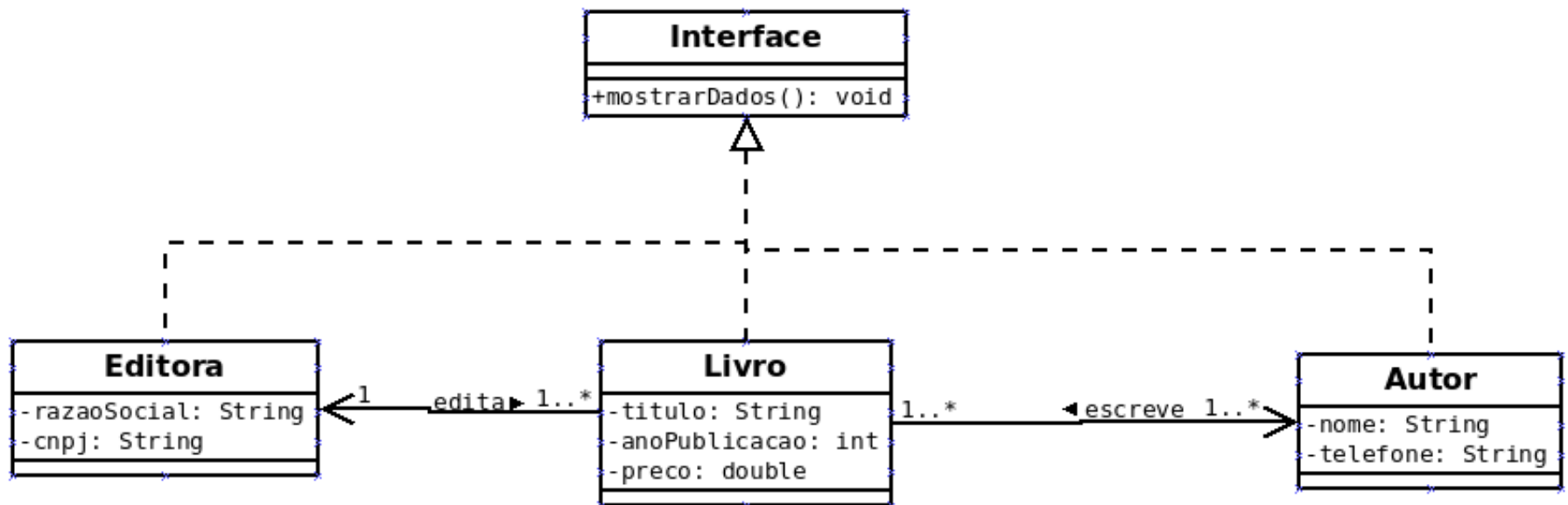
Interface

Interfaces definem um conjunto de comportamentos mandatórios a um grupo de classes, sem a necessidade de uma relação forte como a herança.



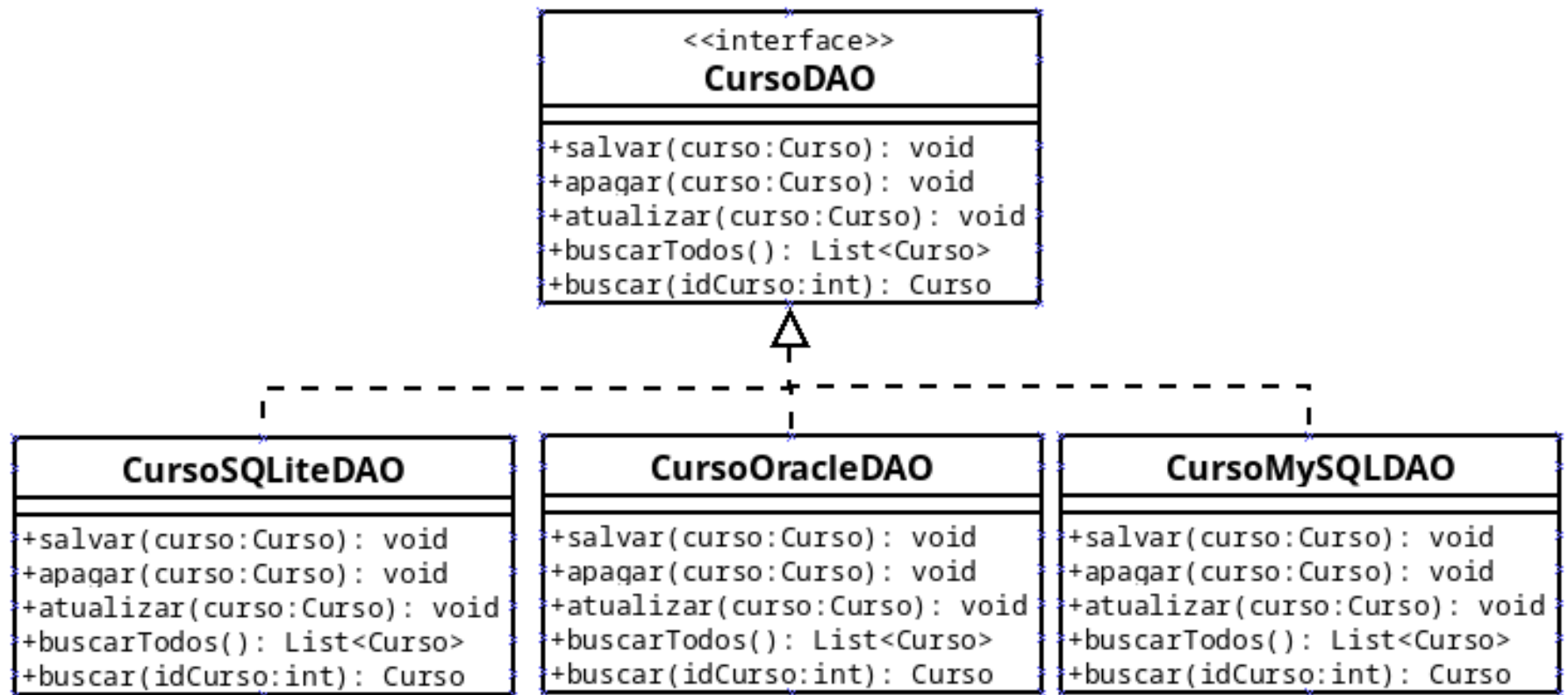
Interface

Outro exemplo



Interface

Outro exemplo



Interface

Interfaces **NÃO** podem ser instanciadas.

Interfaces podem conter:

- Métodos abstratos
- Constantes (*final*)
- Métodos com implementação padrão, desde que indicados pela palavra-chave *default* (Java 8+). Ele pode ser sobrescrito.
- Métodos estáticos (Java 8+)

Implementar uma interface implica em **cumprir um contrato**, se comprometendo a implementar todos os métodos abstratos estabelecidos.

Interface

Definição da interface

```
public interface MinhaInterface {  
    String valor = 15; //implicitamente, é uma constante  
  
    void metodoAbstrato(); //deverá ser sobrescrito na classe  
  
    default void metodoDefault() { // PODE ser sobrescrito nas classes  
        System.out.println("Método Defaultt");  
    }  
    static void metodoStatic() { // NÃO pode ser sobrescrito  
        System.out.println("Método Static (da classe)");  
    }  
}
```


Interface

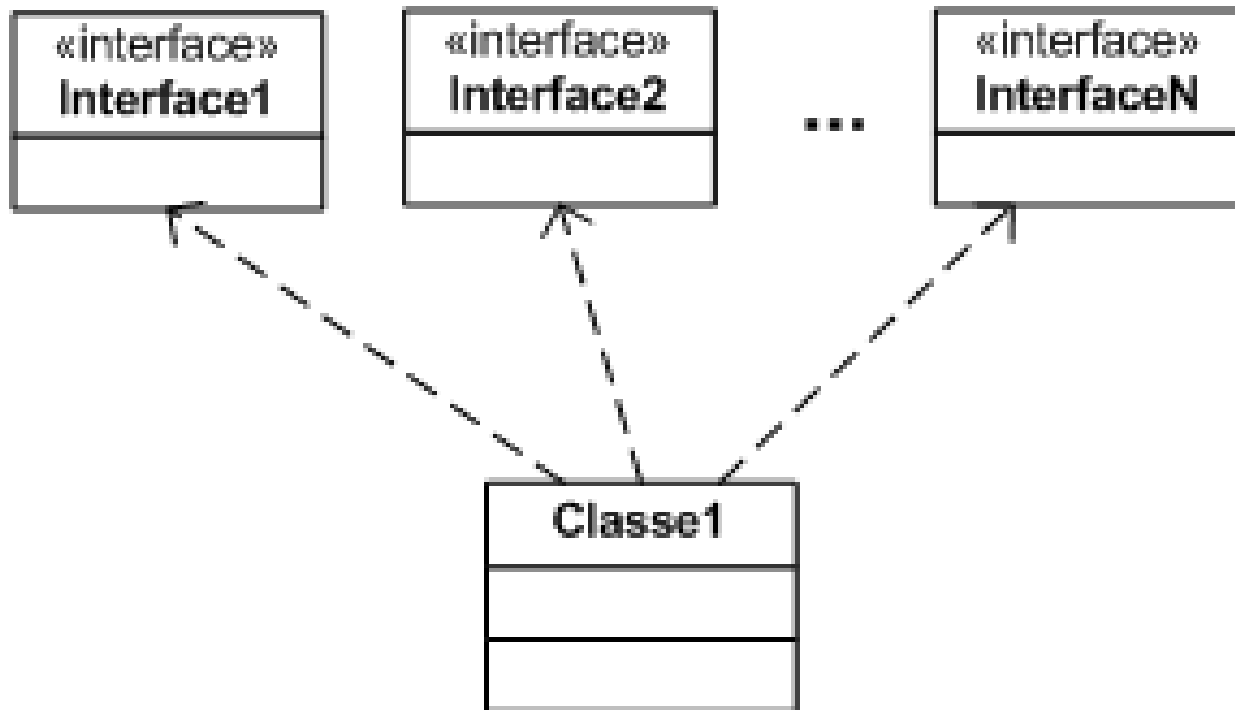
Implementação da interface: palavra reservada **implements**

```
public class MinhaClasse implements MinhaInterface{  
    ...  
  
    @Override  
    public void metodoAbstrato() {  
        //implementação do método abstrato  
    }  
  
    ...  
}
```

Neste exemplo, o método *metodoDefault()* não foi sobrescrito.

Interface

Uma classe pode implementar **múltiplas interfaces**



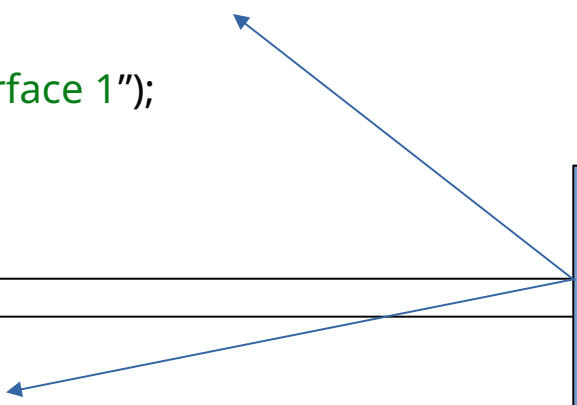
Interface

Implementação de múltiplas interfaces: nomes das interfaces separadas por vírgula

```
public class Classe1 implements Interface1, Interface2, Interface3{  
  
    //Implementação dos métodos abstratos das interfaces utilizadas  
  
}
```

```
public interface Interface1{  
    default void mostrarMensagem() {  
        System.out.println("Mensagem da Interface 1");  
    }  
}
```

**Métodos
default com
mesma
assinatura**



```
public interface Interface2{  
    default void mostrarMensagem() {  
        System.out.println("Mensagem da Interface 2");  
    }  
}
```

```
public class Classe implements Interface1, Interface2{
```

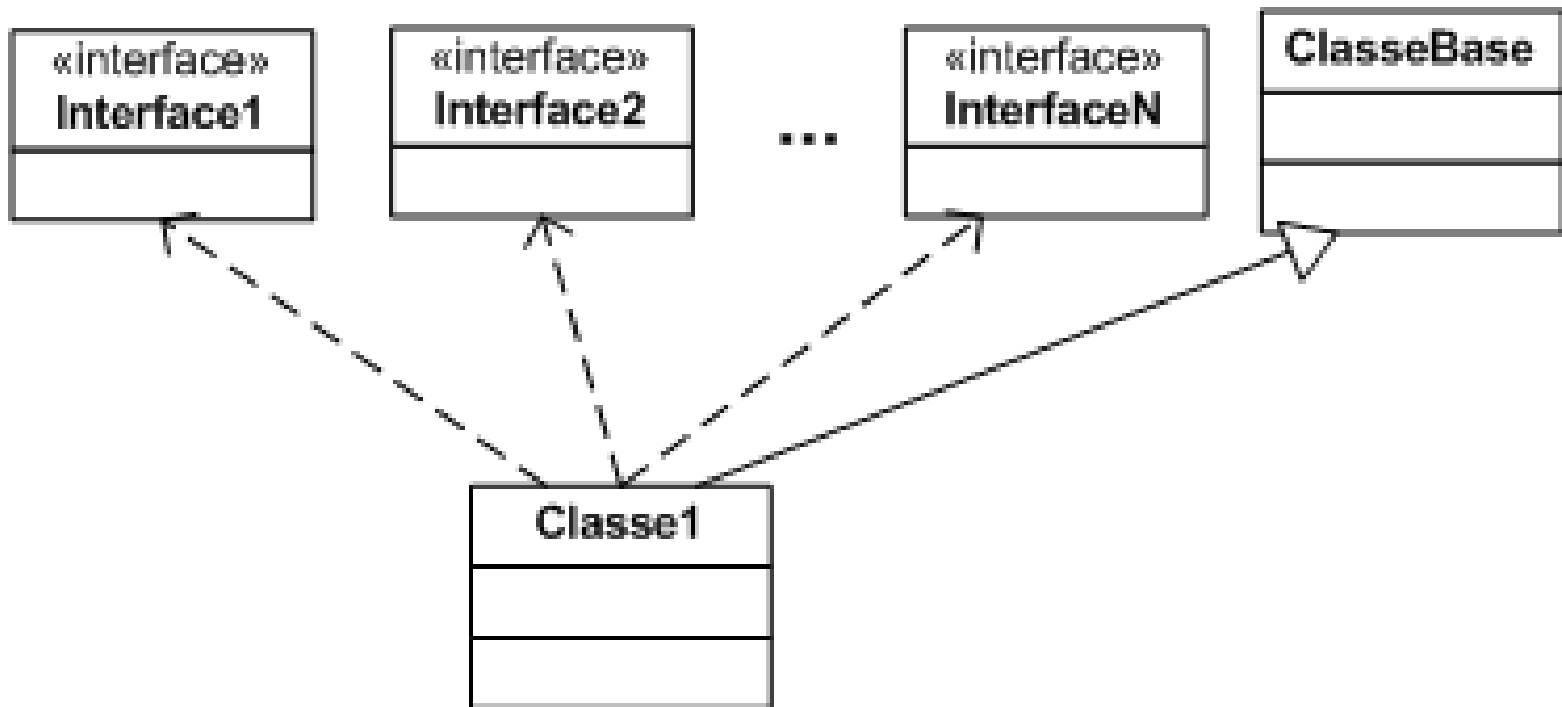
@Override

```
void mostrarMensagem() {  
    Interface1.super.mostrarMensagem();  
    Interface2.super.mostrarMensagem();  
    //Ou poderia mostrar qualquer outra coisa  
}  
}
```

**É preciso
sobrescrever
o método!!**

Interface

Uma classe pode implementar múltiplas interfaces e herdar apenas 1 classe



Interface

Implementação

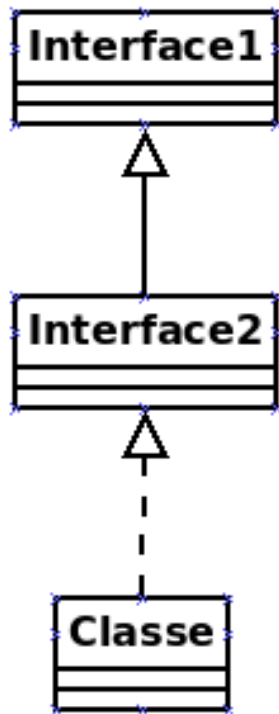
```
public class Classe1 extends ClasseBase implements Interface1,  
Interface2, Interface3{
```

```
    //Implementação dos métodos abstratos das interfaces utilizadas.  
    //Se ClasseBase for abstrata e possuir métodos abstratos, estes  
    //também deverão ser implementados.
```

```
}
```

Interface

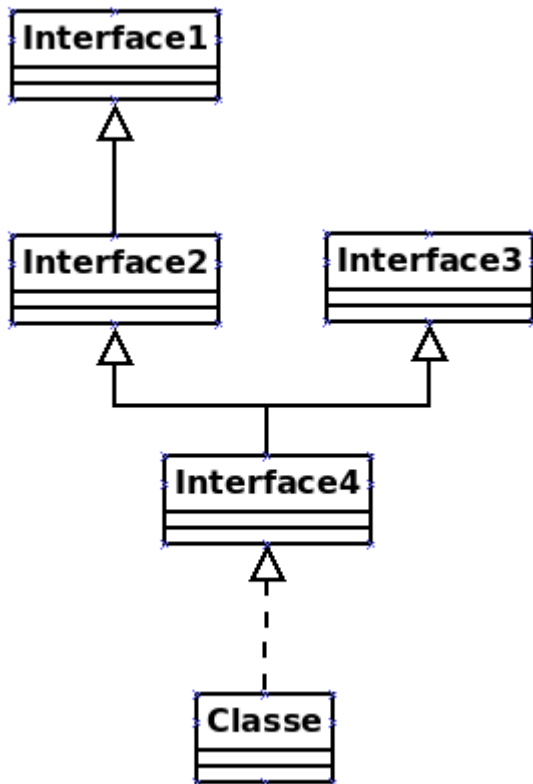
Uma interface pode herdar outras interfaces.



```
public interface Interface2 extends Interface1{  
    ...  
}
```

Interface

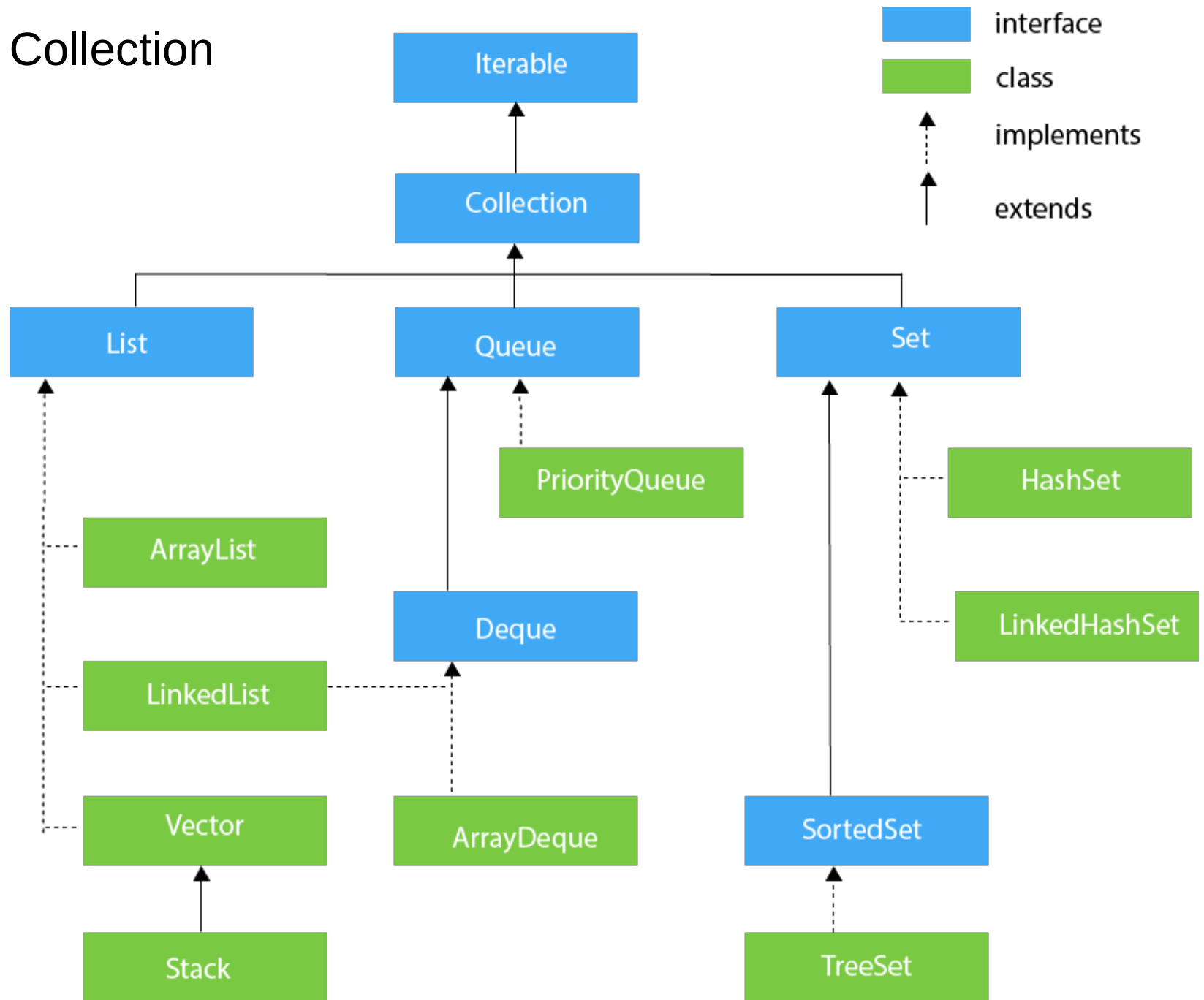
Uma interface pode herdar outras interfaces.



```
public interface Interface2 extends Interface1{  
    ...  
}
```

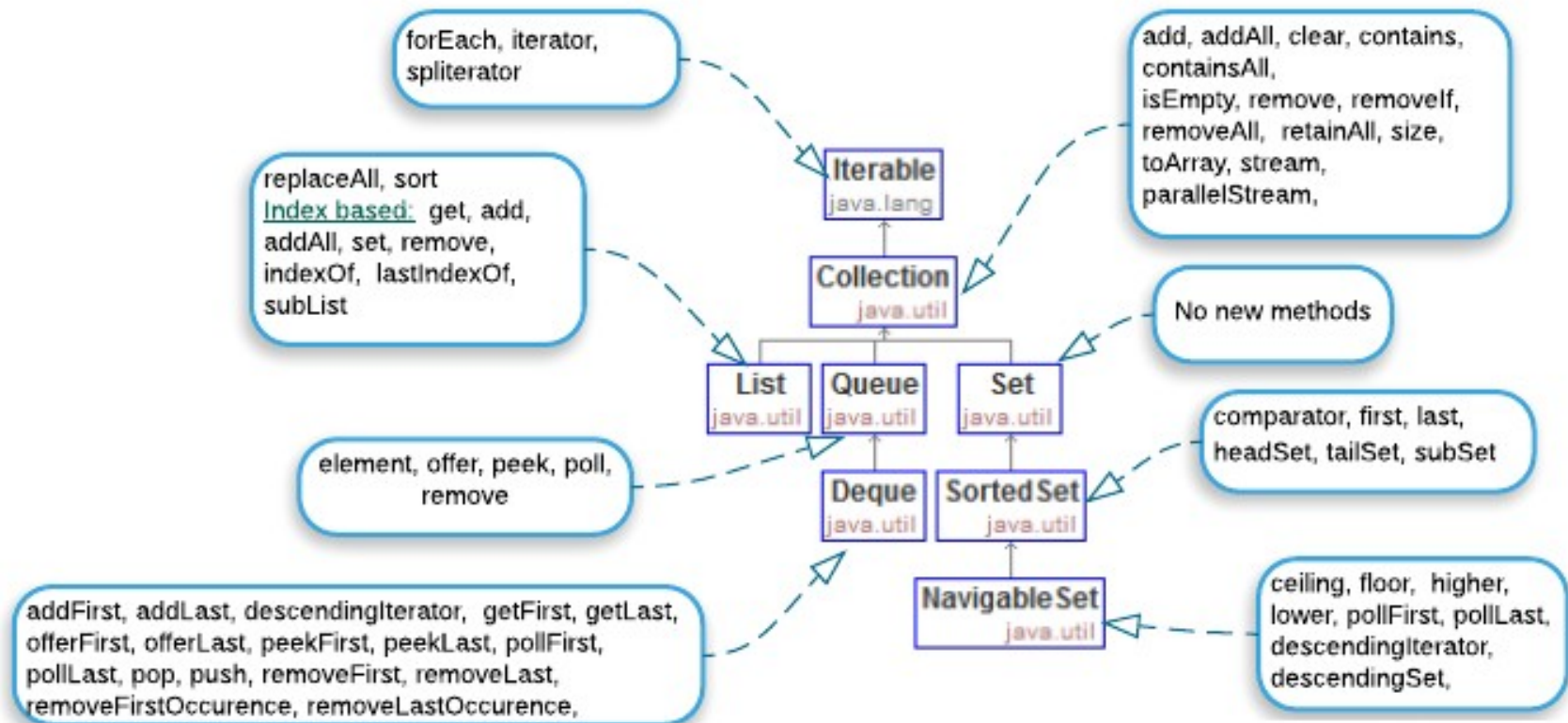
```
public interface Interface4 extends Interface2,  
Interface3{  
    ...  
}
```


API Collection



API Collection

collections Methods:



Exercício I

Implemente em Java o diagrama do slide seguinte.

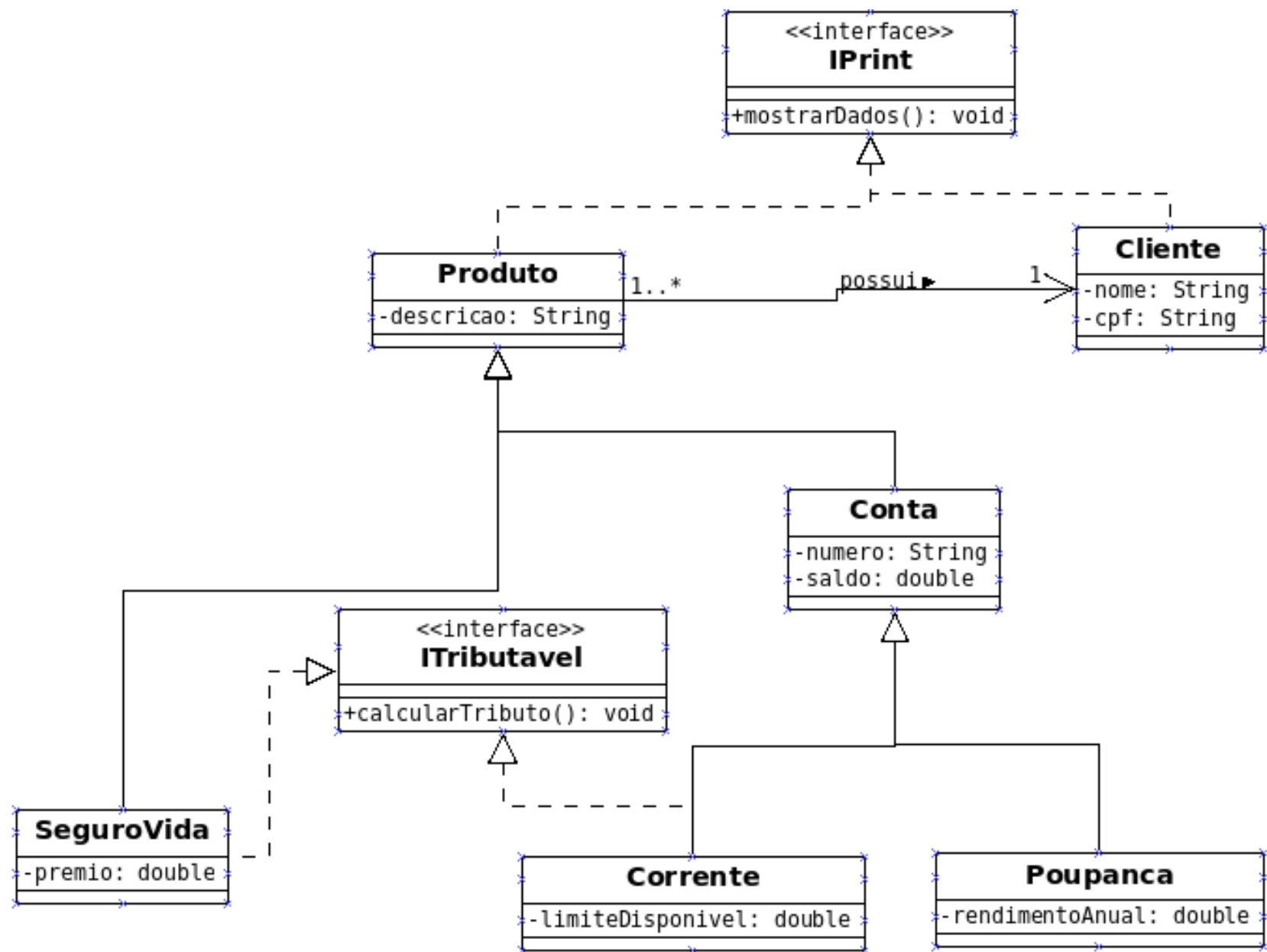
No hipotético sistema bancário, os produtos são tributáveis da seguinte maneira pelo método ***calcularTributo()***:

Conta-corrente: 5% do saldo da conta;

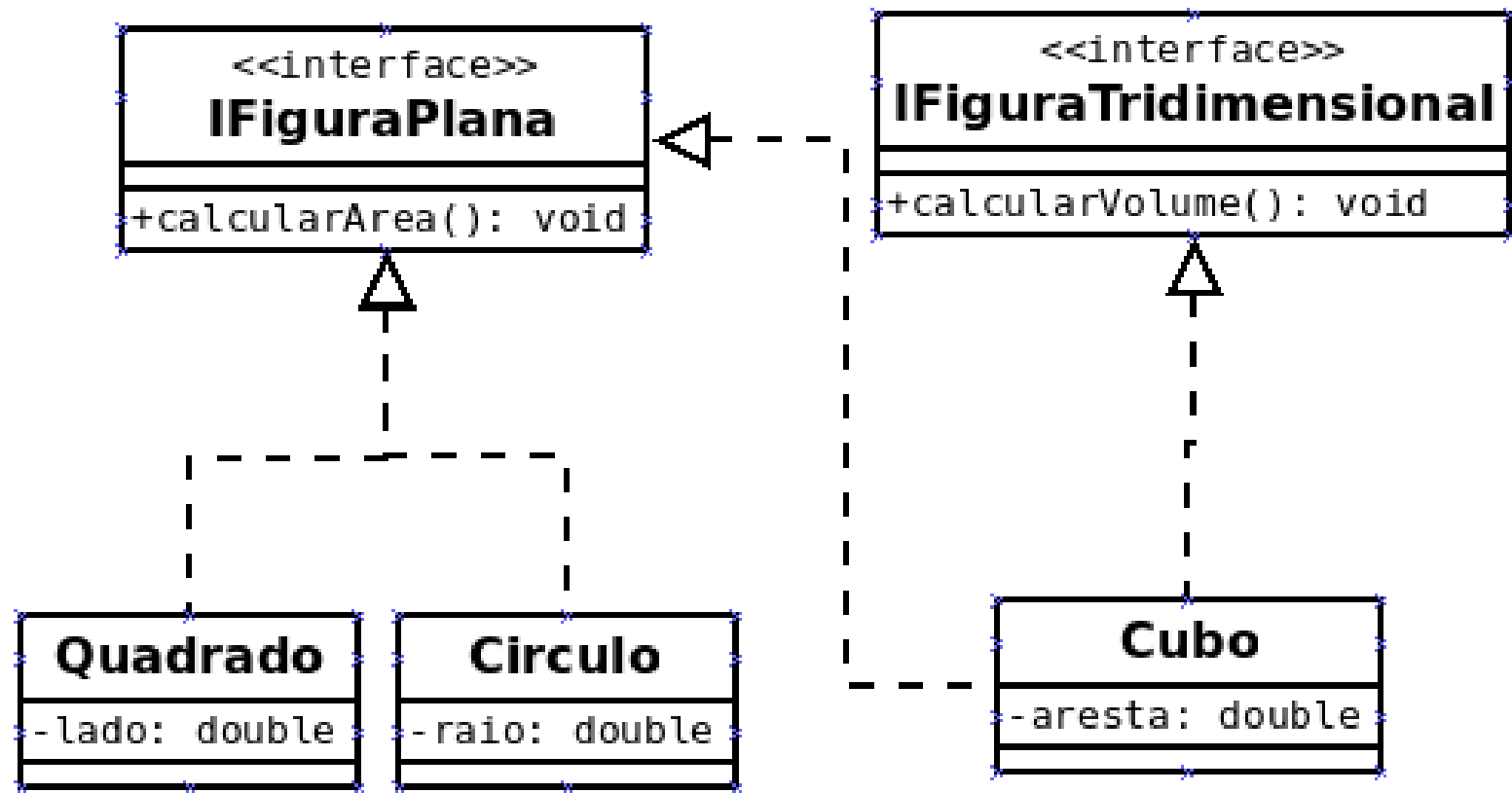
Seguro de vida: 7% do valor do prêmio do seguro

Poupança: não há tributação.

Exercício I



Exercício II



Área do quadrado: lado^2

Área do círculo: $\text{raio}^2 * \text{PI}$ (valor de PI: 3.14)

Área do cubo: $\text{aresta}^2 * 6$

Volume do cubo: aresta^3