

# Programação Orientada a Objetos

## **Classes abstratas**

*slides baseados no cap 9 da apostila FJ11*

Ely – [elydasilvamiranda@gmail.com](mailto:elydasilvamiranda@gmail.com)

## Classes Abstratas

- Há classes que apenas idealizam de forma genérica um tipo:
  - Geralmente não possuem um uso real no sistema a menos que sejam estendidas. Exemplos:
    - Classe genéricas e suas classes derivadas;
    - Funcionario: Secretária, Professor, Gerente, Diretor;
    - Conta: Conta-Corrente, Poupança, Conta-Salário, Conta-Imposto;
  - Dizemos que apesar delas definirem um tipo, elas são apenas rascunhos.

## Classes Abstratas

- Muitas vezes, ao projetar uma classe temos:
  - Boa noção de todos os atributos;
  - Visão de como será a hierarquia de classes;
  - *Porém: há a certeza de que os métodos que podem mudar em cada classe;*

3

## Classes Abstratas

- Problema:
  - Não sabemos como alguns métodos pertencentes a todas as classes devem ser implementados;
  - Implementar esses métodos e esperar que as classes sobrescrevam não é seguro;
- Solução:
  - Definem-se esses métodos na classe raiz da hierarquia como abstratos:
    - Esse métodos não são implementados;
    - A subclasses são obrigadas a implementá-los;

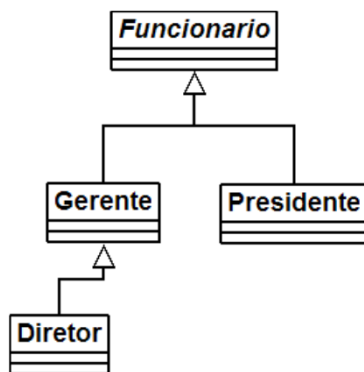
4

## Métodos e classes abstratos

- Métodos abstratos possuem apenas assinatura, ou seja, não possuem implementação;
- Toda classe que possui métodos abstratos também é uma classe abstrata:
  - Não pode ser instanciada;
  - Quem herdar dessa classe deve implementar os métodos abstratos...
  - ... ou permanecer como abstrata.

5

## Hierarquia de Classes



6

## Problemas

- Supondo que a seguinte implementação da classe Funcionário:

```
class Funcionario {  
    protected double salario;  
    public double getBonificacao() {  
        return this.salario * 1.2;  
    }  
    ...  
}
```

- Será que todos os que herdarem de Funcionario terão 20% de bonificação?
- Se não, será que todos os programadores terão a preocupação em sobrescrever o método?

7

## Exemplo de método abstrato

- Usamos a palavra reservada abstract para dizer que a classe e os métodos são abstratos:

```
abstract class Funcionario {  
    protected double salario;  
    public abstract double getBonificacao() ;  
    ...  
}
```

*Apenas a assinatura do métodos. Subclasses devem implementar esse método*

- Se tentarmos instanciar a classe, ocorrerá um erro:  
Funcionario f = new Funcionario(); // não compila

8

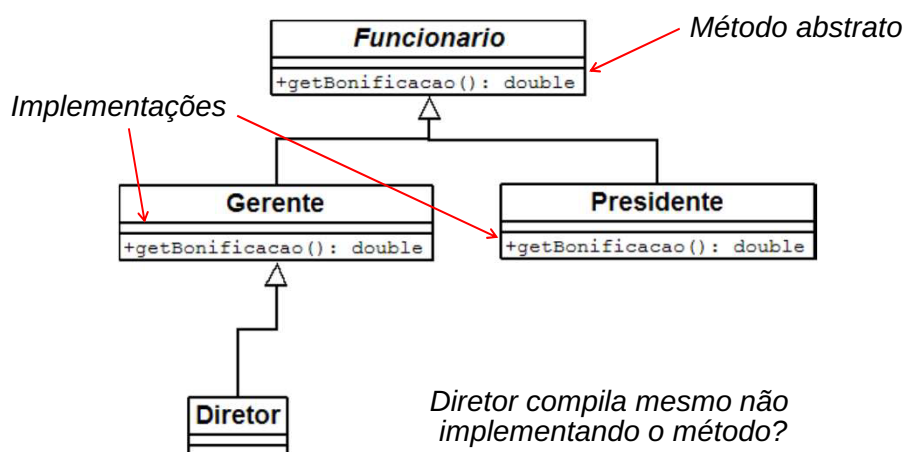
## Extendendo uma classe abstrata

```
class Gerente extends Funcionario {  
    public double getBonificacao() {  
        return this.salario * 1.4 + 1000;  
    }  
}
```

*Como gerente implementou o método abstrato, pode ser usado normalmente no sistema*

9

## Complicando o exemplo 1

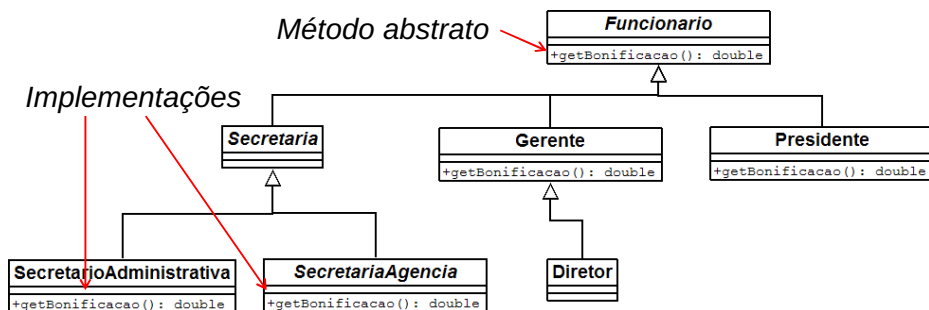


*Diretor compila mesmo não implementando o método?*

Sim, pois ela herda a implementação de Gerente

10

## Complicando o exemplo 2



Secretaria continua abstrata?

SecretariaAdministrativa e SecretariaAgencia compilam?

11

## Outras possíveis combinações

- Uma classe que estende uma classe normal também pode ser abstrata:
  - Ela não poderá ser instanciada, mas sua classe pai sim;
- Uma classe abstrata não precisa necessariamente ter um método abstrato:
  - Alguns frameworks exigem que você instancie suas classes de domínio de classes abstratas do próprio framework;

12

# Programação Orientada a Objetos

**Classes abstratas**

*slides baseados no cap 9 da apostila FJ11*

---

Ely – [elydasilvamiranda@gmail.com](mailto:elydasilvamiranda@gmail.com)