



**UNIFACS**  
LAUREATE INTERNATIONAL UNIVERSITIES

# Programação Orientada a Objetos

Aulas nº X: Classes Abstratas

Prof. Luis Gustavo Araujo  
2018

# Objetivo

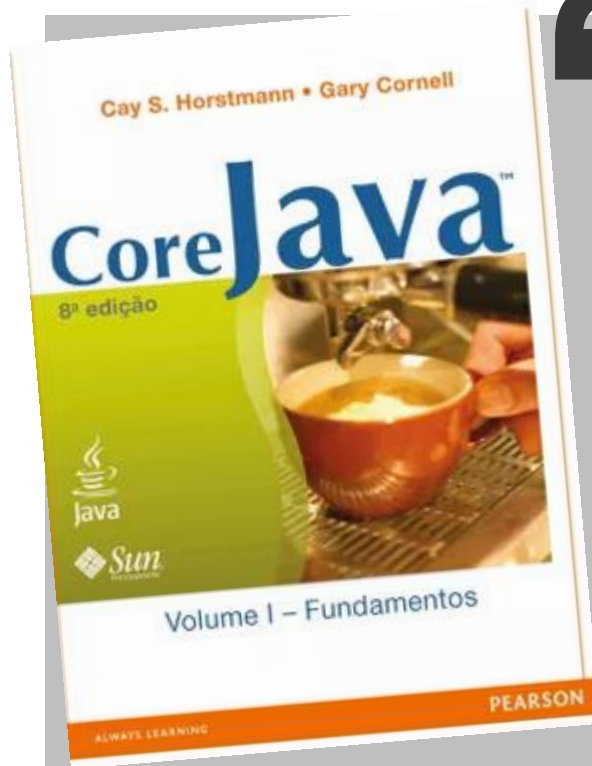
Compreender o conceito e funcionamento de  
Classes Abstratas

Classes

Abstratas

POO

# Classes Abstratas



“

À medida que você sobe na hierarquia de herança, as classes tornam-se mais gerais, e provavelmente mais abstratas. Em algum momento, a classe torna-se tão geral que você pensaria nela mais como uma base para as outras classes do que como uma classe com instâncias específicas que você quer utilizar.

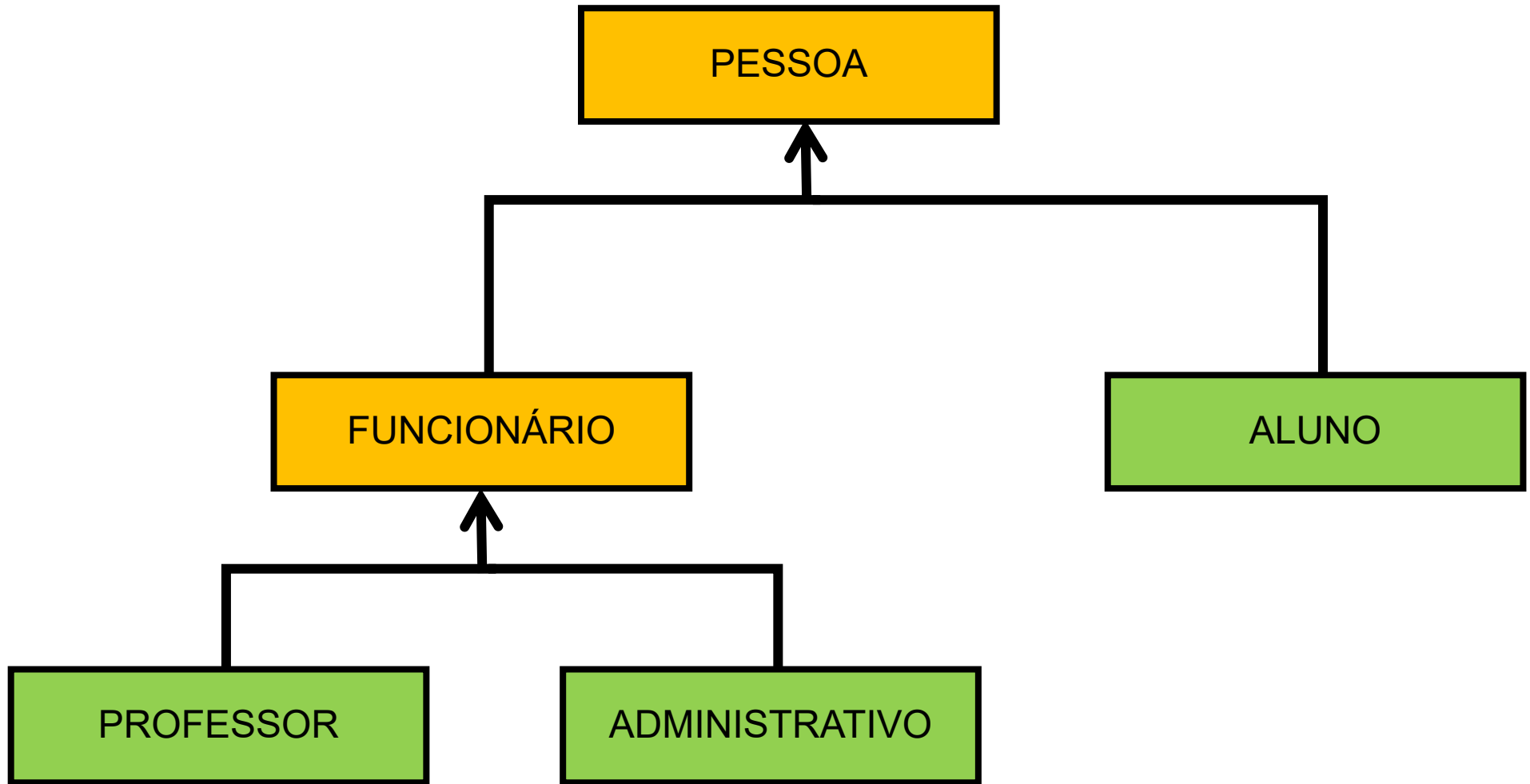
***Horstmann  
& Cornell***

Core Java

# Classes Abstratas

Dado os níveis de herança que criamos, em alguns códigos, a classe torna-se tão geral que não usamos ela como instâncias de objeto. Como exemplo, temos a classe Pessoa (*base para Funcionario e Aluno*).

# Classes Abstratas x Concretas



Classes Abstratas



Classes Concretas

# Classes Abstratas

Desbravando Java e  
Orientação a Objetos  
Um guia para o iniciante da linguagem



Casa do  
Código

RODRIGO TURINI

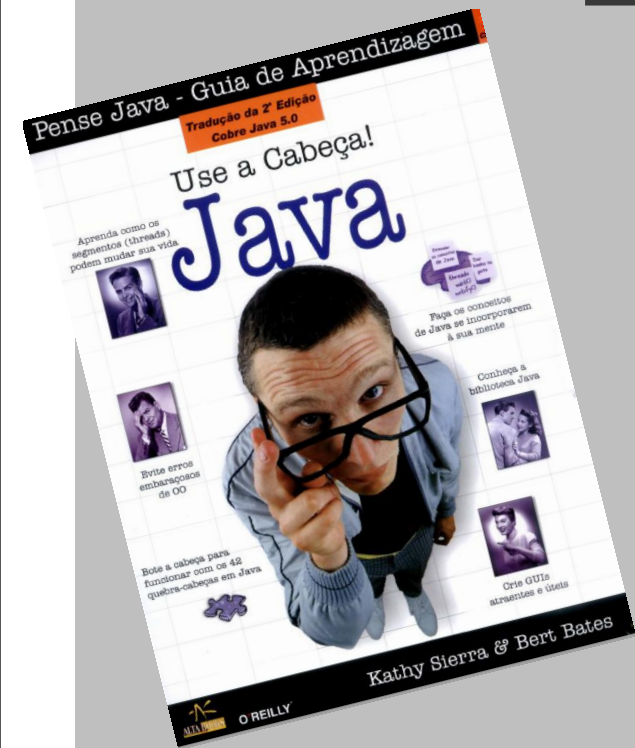
*A partir do momento em que tornamos nossa classe abstrata, o compilador vai impedir que qualquer código tente instanciar uma Pessoa.*

**Rodrigo  
Turini**

Desbravando Java e Orientação a Objetos\*

# Classes Abstratas

“



*Tentar criar um Aluno é como **ter um pesadelo em que acontece um acidente no teletransporte de Jornadas nas Estrelas**. Aquele em que em algum lugar do processo de teletransporte algo errado acontece no buffer.*

**Seirra  
& Bates**

Use a Cabeça! Java\*

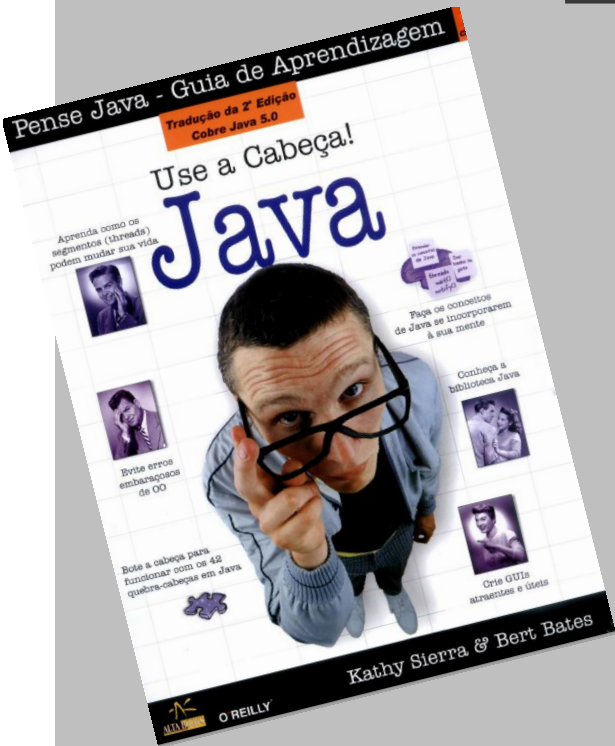
# Classes Abstratas

“

*Felizmente, há uma maneira simples de impedir que uma classe seja instanciada. Em outras palavras, de impedir que alguém use “new” com esse tipo. Se marcarmos a classe com **abstract**, o compilador impedirá que qualquer código, esteja onde estiver, crie uma instância desse tipo.*

**Seirra  
& Bates**

Use a Cabeça! Java\*





# Classes Abstratas

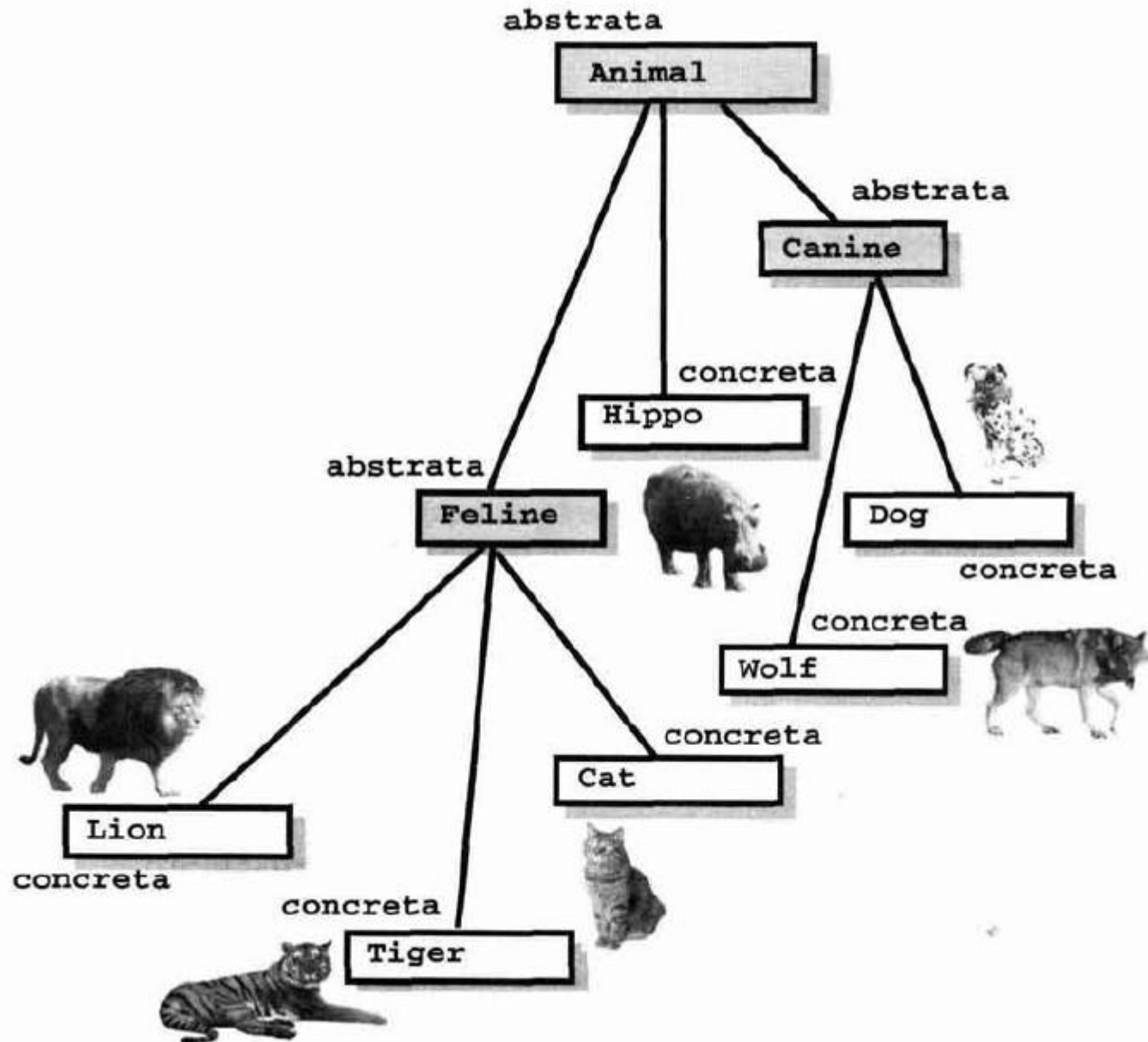


Figura 1.0

# De classe Concreta...

```
1. public class Funcionario{  
2.     private String nome;  
3.     private double salario;  
4.  
5.     public Funcionario(String nome, double salario){  
6.         this.nome = nome;  
7.         this.salario = salario;  
8.     }  
9. }
```

```
1. Funcionario f = new Funcionario(); //OK
```

# Para classe Abstrata.

```
1. abstract public class Funcionario{  
2.     private String nome;  
3.     private double salario;  
4.  
5.     public Funcionario(String nome, double salario){  
6.         this.nome = nome;  
7.         this.salario = salario;  
8.     }  
9. }
```

```
1. Funcionario f = new Funcionario(); //ERRO
```

# Observação

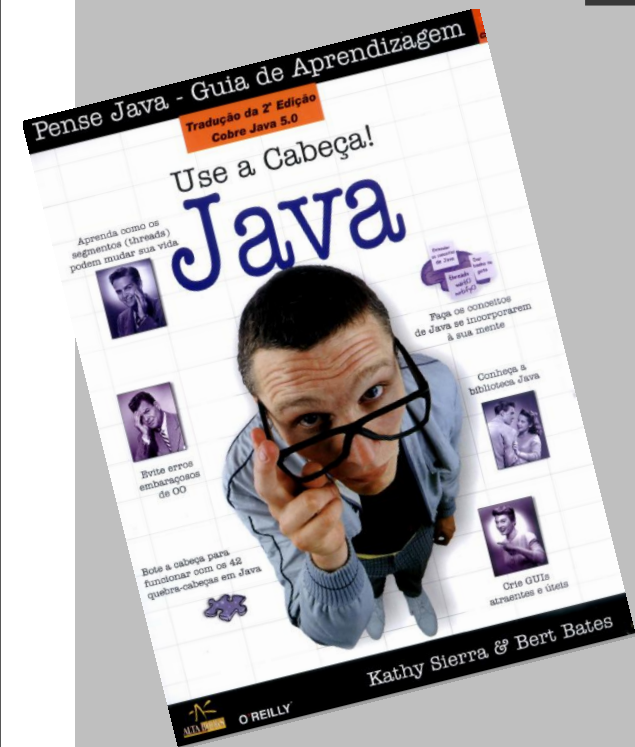
Embora as classes abstratas não possam ser instanciadas, elas podem ser usadas como referências de objetos das classes filhas (subclasses), para fins de polimorfismo!

# Testando Classes Abstratas

1. Pessoa p = **new** Pessoa(); //ERRO
- 2.
3. Pessoa a = **new** Aluno(); //OK
- 4.
5. Funcionario f = **new** Funcionario(); //ERRO
- 6.
7. Funcionario g = **new** Gerente(); //OK
- 8.
9. Funcionario ad = **new** Administrativo(); //OK
- 10.
11. Pessoa g2 = **new** Gerente(); //OK
- 12.
13. Pessoa ad2 = **new** Administrativo(); //OK

# Em resumo

“



*Uma classe abstrata praticamente\* não tem utilidade, valor, razão de existir, ao menos que seja **estendida**.*

# Seirra & Bates

Use a Cabeça! Java

# Observação

\*Uma classe abstrata pode conter atributos e métodos estáticos, então ela pode ser útil mesmo que não sendo pai de nenhuma classe.

# Referências Técnicas

Sierra, Kathy, and Bert Bates. **Use a cabeça!: java**. Alta Books, 2007.

HORSTMANN, S.; CORNELL, G.; **Core JAVA**. 1. vol. São Paulo: Pearson Education, 2010.

TURNI, R. **Desbravando Java e Orientação a Objetos**: um guia para o iniciante da linguagem. São Paulo: Casa do Código, 2016.





**UNIFACS**  
LAUREATE INTERNATIONAL UNIVERSITIES

# Programação Orientada a Objetos

Aulas nº X: Classes Abstratas

Prof. Luis Gustavo Araujo  
2018