

Linguagem de Programação Orientada a Objetos

Aula 07 – Classes Abstratas

Ms Gustavo Molina

<msc.gustavo.unip@gmail.com>

Classes Abstratas - Conceito

- ✓ Classes abstratas em Java são classes que não serão instanciadas, ou seja, são classes em que não criaremos nenhum objeto dela, diretamente. Dessa forma podemos dizer que em Java criamos Superclasses (Classes Mãe) Abstratas.



Classes Abstratas - Conceito

- ✓ Usamos classes abstratas para representar grupos que tem características comum, mas que em alguns detalhes específicos, agem de maneira diferente.
- ✓ As classes abstratas estão relacionadas com polimorfismo.



Classes Abstratas – Exemplos Práticos

1. Se você for a uma concessionária para comprar um veículo, você nunca irá escolher um "Carro", você escolhe um Gol, um Fiat, enfim, um modelo específico.
2. Se você for dono de um empresa e for contratar alguém, você não contrata simplesmente um "Funcionário", você contrata uma secretária, um técnico de TI, enfim, você contrata alguém com uma profissão específica para fazer ma função específica.
3. Se você for jogar o jogo Colheita Feliz, você não vai colocar um "Animal" no jogo, você coloca uma vaca, um cão ou uma ovelha. Ou seja, um animal específico.

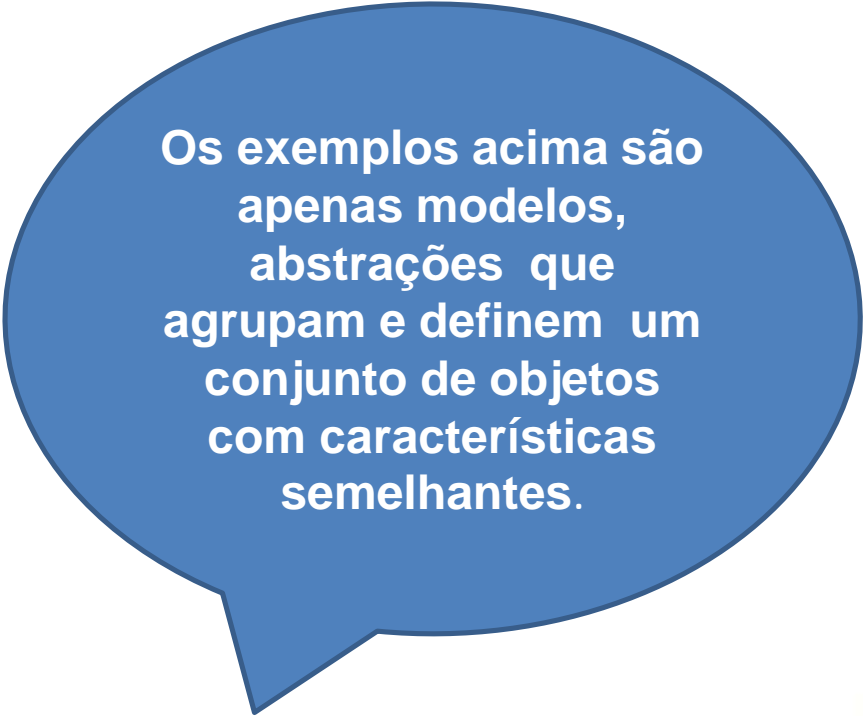
Classes Abstratas – Exemplos Práticos

- ❖ Na verdade, em nosso mundo real, só existem objetos. As classes abstratas, servem para classificar grupos de objetos.
- 1. Por exemplo, "Carro" é uma classe abstrata que reúne um conjunto de características que classificam os carros: potência do motor, possuem pneus, possuem portas, se locomovem, entre outros.
- 2. "Funcionário" é uma classe abstrata que reúne um conjunto de características, gerais, de um funcionário: tem salário, uma missão na empresa, pagam impostos, tem um chefe, entre outros.



Classes Abstratas – Exemplos Práticos

- ❖ No mundo real não existe o objeto “Carro” e sim fusca ou Ferrari.
- ❖ Ninguém pede carne de “Animal” na churrascaria, e sim de boi ou carneiro.
- ❖ Ninguém se candidata a uma vaga para trabalhar como “Funcionário”.

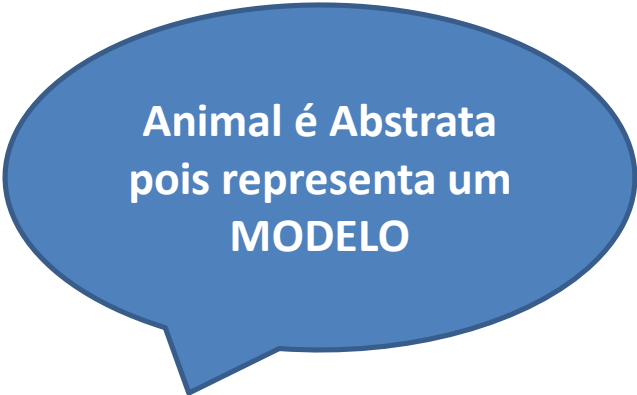


Os exemplos acima são apenas modelos, abstrações que agrupam e definem um conjunto de objetos com características semelhantes.

Classes Abstratas – Na prática

-->Animal.java

```
public abstract class Animal {  
    protected String nome;  
    protected int numeroPatas;  
  
    public abstract void som();  
  
}
```



Animal é Abstrata
pois representa um
MODELO

Classes Abstratas – Na prática

-->Vaca.java

```
public class Vaca extends Animal {  
    public Vaca(){  
        this.nome = "Vaca";  
        this.numeroPatas = 4;  
    }  
  
    @Override  
    public void som(){  
        System.out.println("A " + this.nome + " que tem " + this.numeroPatas + " patas, faz MUUUU");  
    }  
}
```


Classes Abstratas – Na prática

-->Gato.java

```
public class Gato extends Animal{
    public Gato(){
        this.nome = "Gato";
        this.numeroPatas = 4;
    }

    @Override
    public void som(){
        System.out.println("O " + this.nome + " que tem " + this.numeroPatas + " patas, faz MIAU");
    }
}
```

Classes Abstratas – Na prática

-->Carneiro.java

```
public class Carneiro extends Animal{
    public Carneiro(){
        this.nome = "Carneiro";
        this.numeroPatas = 4;

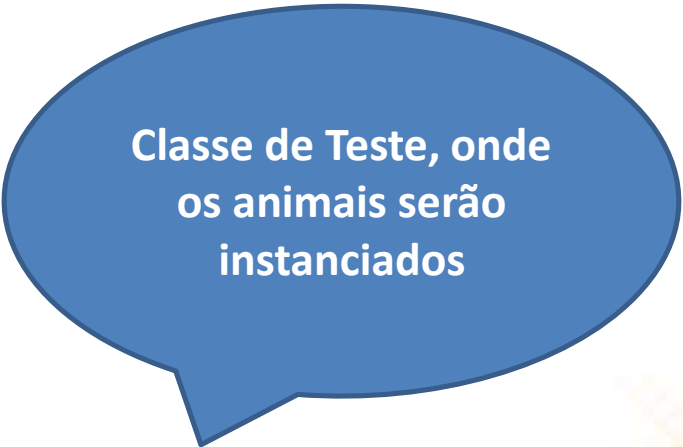
    }

    @Override
    public void som(){
        System.out.println("O " + this.nome + " que tem " + this.numeroPatas + " patas, faz BÉÉÉ");
    }
}
```

Classes Abstratas – Na prática

-->Bichos.java

```
public class Bichos {  
  
    public static void main(String[] args) {  
        Vaca mimosa = new Vaca();  
        Gato bichano = new Gato();  
        Carneiro barnabe = new Carneiro();  
  
        mimosa.som();  
        bichano.som();  
        barnabe.som();  
    }  
}
```



Classe de Teste, onde
os animais serão
instanciados

Dúvidas???

