Aula: Classes Abstratas e Interfaces

Instituto Metrópole Digital - UFRN
Disciplina: LP2
Docente: Emerson Alencar

Classes Abstratas

- Usadas quando não faz sentido termos instâncias de determinadas classes;
  - Quando não queremos que determinadas classes que escrevemos não tenham instâncias;
- Serve para manter a consistência do programa

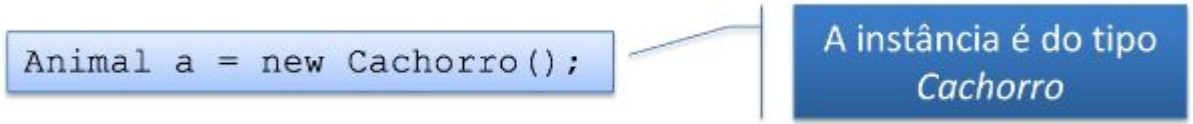
Como definir uma Classe Abstrata

Utilizando o modificador abstract na declaração da classe

```
public abstract class Animal {
    ...
}
```

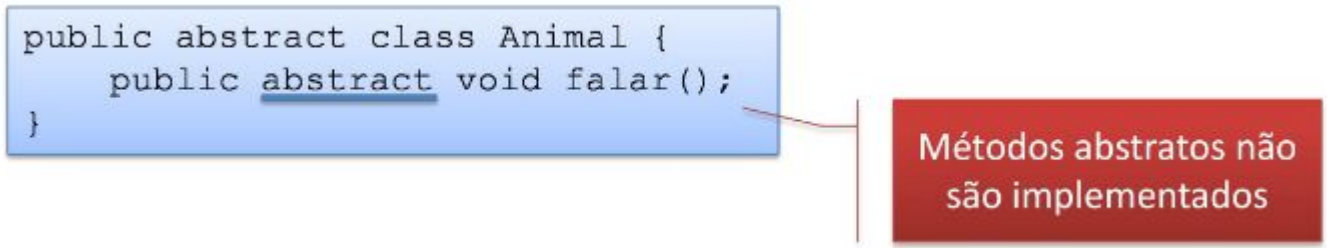
Como usar uma Classe Abstrata?

Criando referência à classe

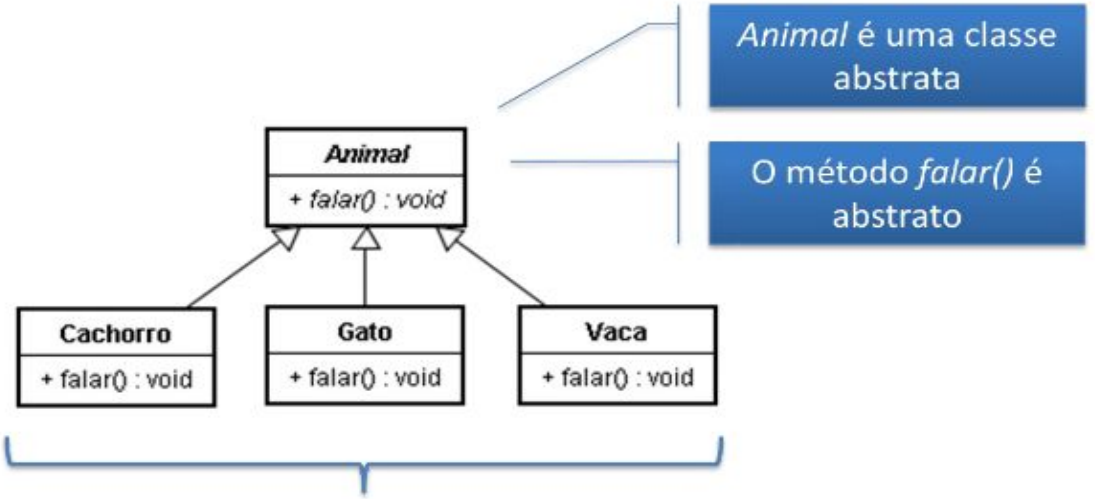


Métodos Abstratos

- Utilizados quando não faz sentido termos a implementação do método em determinada classe;
- Para declarar um método abstrato, basta utilizar o modificador abstract na declaração no método;
- É implementado pelas classes que herdam da classe que o método pertence.



Exemplo:



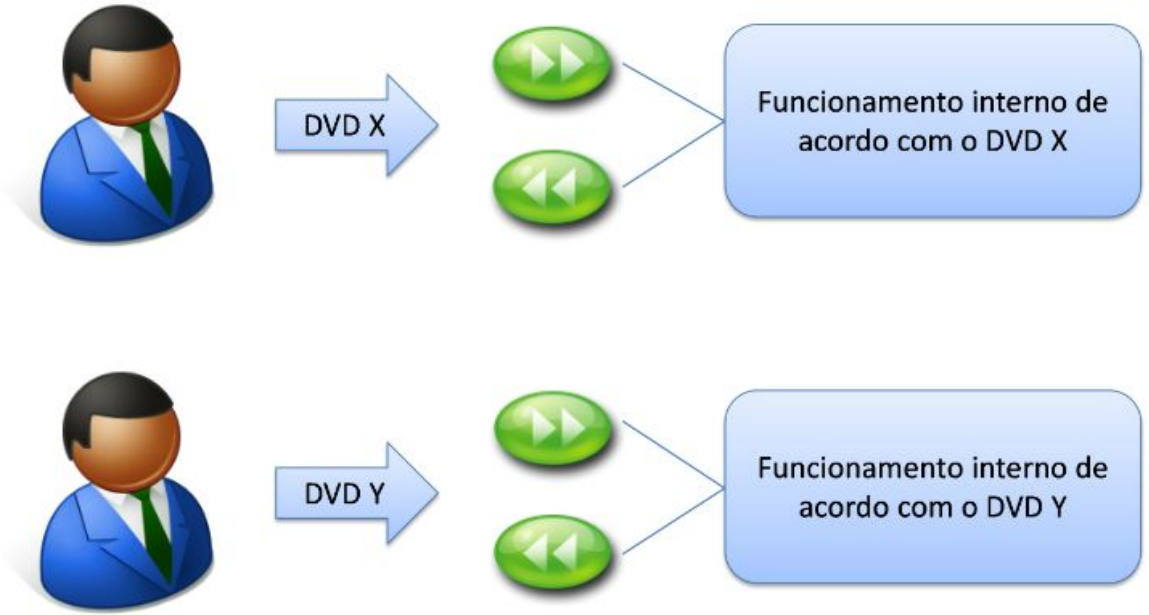
Todas as classes não-abstratas que herdam de uma classe abstrata são obrigadas a implementar os métodos abstratos

Os métodos chamados correspondem aos métodos implementados nas subclasses

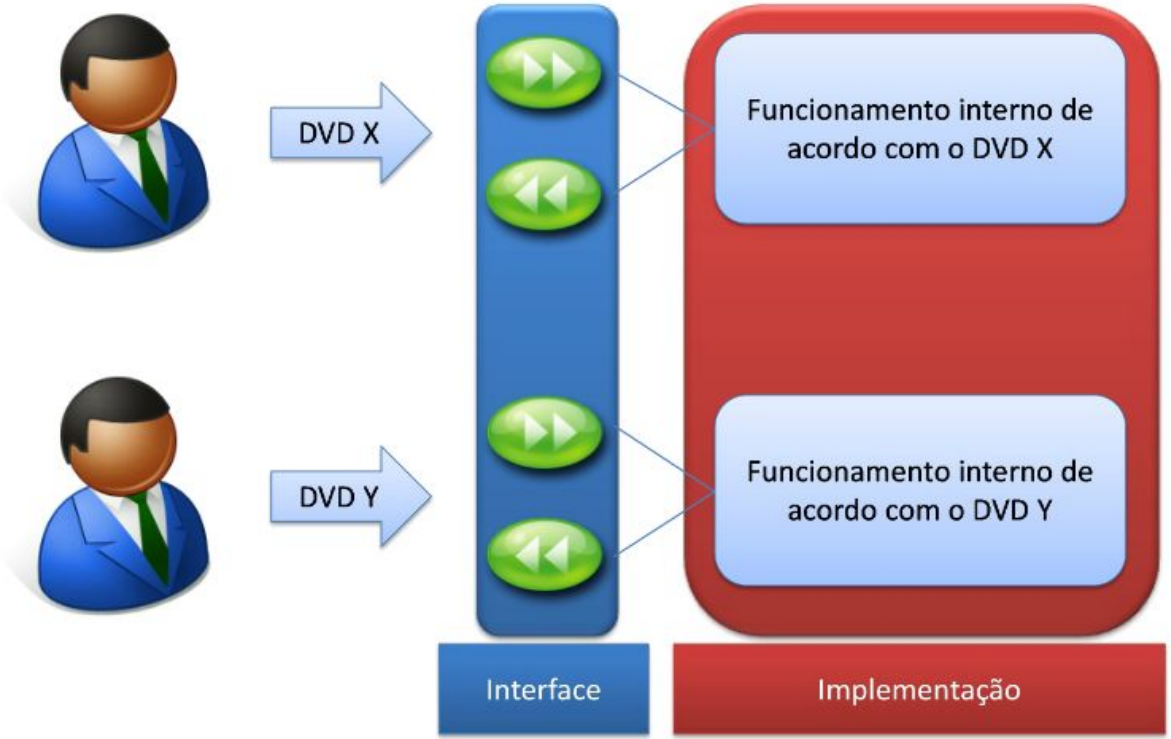
Informações Importantes

- Classes Abstratas não precisam obrigatoriamente ter métodos abstratos
- Métodos Abstratos só podem existir em Classes Abstratas

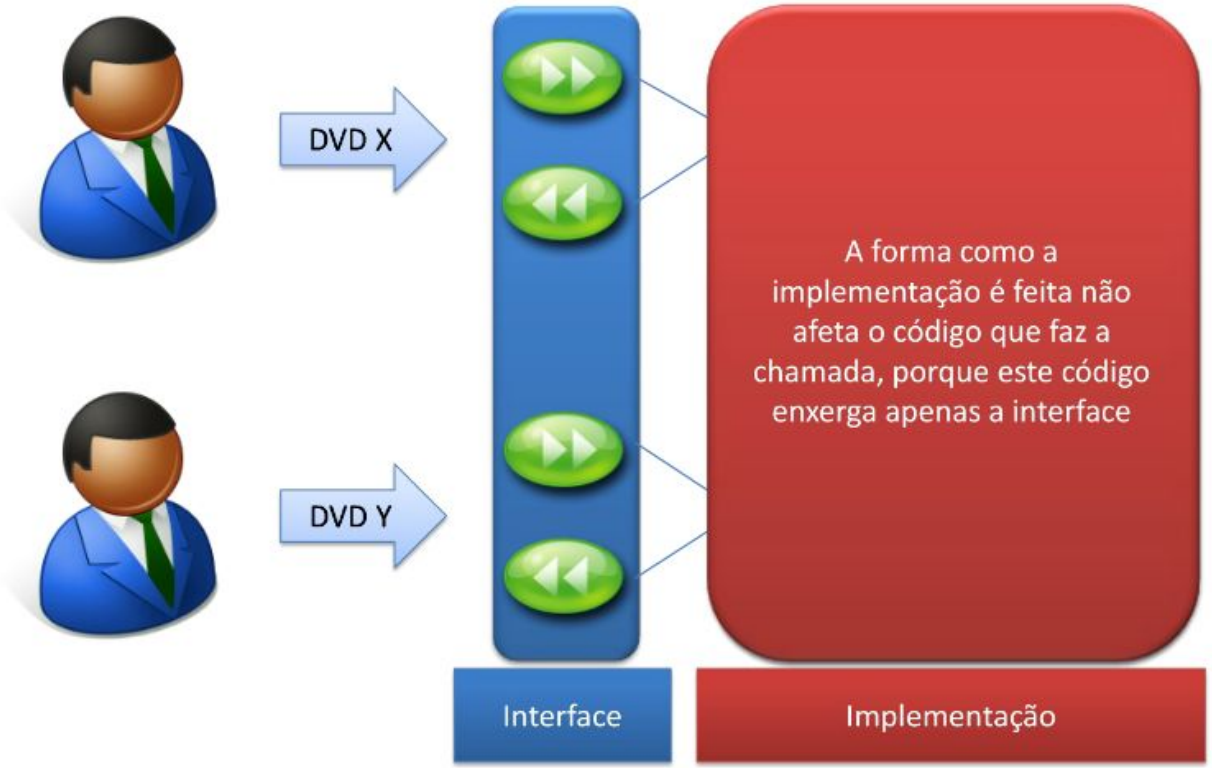
Interface: Um exemplo prático



Interface: Um exemplo prático



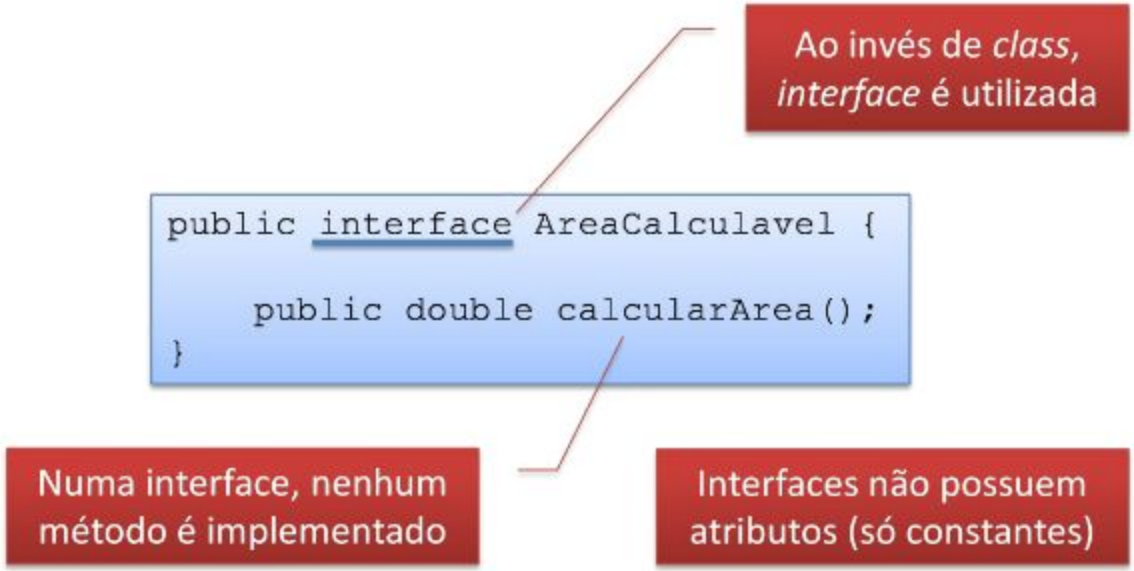
Interface: Um exemplo prático



Interface

- Define os métodos, mas não os implementa;
  - Com exceção dos métodos que usam os modificadores default e static
- A implementação é de responsabilidade de quem implementa a interface;
- O foco é no que o objeto faz, e não em como ele faz
- Interface possibilita mudança de implementação muita mais facilmente, pois quem chama o método não conhece a sua implementação

Declarando Interface



Implementando Interface

```
public class Quadrado implements AreaCalculavel {

    private double lado;

    public double calcularArea() {
        return lado * lado;
    }
}
```

```
public class Circulo implements AreaCalculavel {

    private double raio;

    public double calcularArea() {
        return Math.PI * raio * raio;
    }
}
```

Os métodos da interface são implementados pela classe

Exemplo:

```
AreaCalculavel a = new Quadrado();
a.calcularArea();
```

```
AreaCalculavel a = new Circulo();
a.calcularArea();
```

Utilização de polimorfismo

Informações importantes

- Interfaces podem estender outras interfaces;
- Classes podem estender outra classe, mas apenas podem implementar interfaces;
- Uma classe pode implementar uma ou mais interfaces.

Métodos Default

- Uma Interface pode definir métodos com o modificador default;
- Neste caso, o método é implementado diretamente na Interface

Definindo Métodos Default

```
public interface Calculator {

    double calculate();

    default double calculatePow(double x, int y) {
        return Math.pow(x, y);
    }
}
```

O método default é implementado na interface

```
public class MyCalculator implements Calculator {
    public double calculate() {
        //...
    }
}
```

```
MyCalculator my = new MyCalculator();
double x = my.calculatePow(10, 3);
```

Funciona como um método qualquer

Métodos Estáticos

- Interfaces também podem implementar métodos definidos com o modificador static;
- O método é acessível diretamente pela interface, sem precisar que ocorra a criação de objeto’s.

Definindo Métodos Estáticos

```
public interface Calculator {

    double calculate();

    static double calculatePow(double x, int y) {
        return Math.pow(x, y);
    }
}
```

O método estático é implementado na interface

```
Calculator.calculatePow(10, 3);
```

Método chamado diretamente na interface *Calculator*

Classes Abstratas ou Interfaces?

- A escolha entre classes abstratas ou interfaces tem dois aspectos que precisam ser levados em consideração:
  - Conceitual
    - Classes abstratas são classes que não podem ter instâncias
    - Interfaces determinam como um objeto será exposto
  - Prático
    - Uma classe pode implementar mais de uma interface
    - Uma classe abstrata pode ter atributos
- Importante: Classes Abstratas e interfaces têm o objetivo comum de favorecer o uso do polimorfismo