Abstração (ciência da computação)

Origem: Wikipédia, a enciclopédia livre.

Abstração é a habilidade de concentrar nos aspectos essenciais de um contexto qualquer, ignorando características menos importantes ou acidentais. Em modelagem orientada a objetos, uma classe é uma abstração de entidades existentes no domínio do sistema de software.

Por exemplo, imaginamos a abstração referente a classe *Animais*. Há várias entidades na classe *Animais* como *Anfíbios*, *Répteis* e *Mamíferos* que são também sub-classes da classe *Animais*, onde há objetos que contêm cada sub-classe como *Ser-humano*, *Jacaré* e outros.

Uma **classe abstrata** é desenvolvida para representar entidades e conceitos abstratos. A classe abstrata é sempre uma superclasse que não possui instâncias. Ela define um modelo (*template*) para uma funcionalidade e fornece uma implementação incompleta - a parte genérica dessa funcionalidade - que é compartilhada por um grupo de classes derivadas. Cada uma das classes derivadas, completa a funcionalidade da classe abstrata adicionando um comportamento específico.

Exemplo prático da classe

No exemplo abaixo, implementado em Java, a classe abstrata Eletrodomestico a qual será instanciada por outra classe filha:

```java
public abstract class Eletrodomestico {
    private boolean ligado;
    private int voltagem;

    // métodos abstratos //
    /*
     * não possuem corpo, da mesma forma que
     * as assinaturas de método de uma interface
     */
    public abstract void ligar();
    public abstract void desligar();

    // método construtor //
    /*
     * Classes Abstratas também podem ter métodos construtores,
     * porém, não podem ser usados para instanciar um objeto diretamente
     */
    public Eletrodomestico(boolean ligado, int voltagem) {
        this.setLigado(ligado);
        this.setVoltagem(voltagem);
    }

    // métodos concretos
    /*
     * Uma classe abstrata pode possuir métodos não abstratos
     */
    public void setVoltagem(int voltagem) {
        this.voltagem = voltagem;
    }

    public int getVoltagem() {
        return this.voltagem;
    }

    public void setLigado(boolean ligado) {
        this.ligado = ligado;
    }

    public boolean isLigado() {
        return ligado;
    }
}
```

A classe filha:

```java
public class Radio extends Eletrodomestico {

    //atributos...
    public static final short AM = 1;
    public static final short FM = 2;
    private int banda;
    private float sintonia;
    private int volume;

    //metodos do classe Radio...



    //metodo construtor...
    public Radio(int voltagem) {
        super(true, voltagem);
        setBanda(Radio.FM);
        setSintonia(0);
        setVolume(0);
    }

    /**
     * @return the banda
     */
    public int getBanda() {
        return banda;
    }

    /**
     * @param banda the banda to set
     */
    public void setBanda(int banda) {
        this.banda = banda;
    }

    /**
     * @return the sintonia
     */
    public float getSintonia() {
        return sintonia;
    }

    /**
     * @param sintonia the sintonia to set
     */
    public void setSintonia(float sintonia) {
        this.sintonia = sintonia;
    }

    /**
     * @return the volume
     */
    public int getVolume() {
        return volume;
    }

    /**
     * @param volume the volume to set
     */
    public void setVolume(int volume) {
        this.volume = volume;
    }


    /* implementação dos métodos abstratos */
    public void desligar() {
        super.setLigado(false);
        setSintonia(0);
        setVolume(0);
    }

    public void ligar() {
        super.setLigado(true);
        setSintonia(88.1f);
        setVolume(25);
    }

    // abaixo teríamos todos os métodos construtores get e set...

}
```

A classe de Aplicação:

```java
package aula_abstrata;

public class aplicacaoAbstrata {

    /**
     * @param args
     */
    public static void main(String[] args) {
        //Instancia a classe Radio
        Radio radio1 = new Radio(110);

        /*
         * chamando os métodos abstratos implementados
         * dentro de cada classe ( Radio)
         */

        radio1.ligar();
        System.out.print("e o Rádio está ");
        System.out.println(radio1.isLigado() ? "ligado." : "desligado.");

    }

}
```

Ligações externas

- Abstract Methods and Classes (http://docs.oracle.com/javase/tutorial/java/IandI/abstract.html) (em inglês)
