

Aula #02 19/08/2022

Encapsulamento

Programação Orientada a Objetos

Prof. Carlos Veríssimo¹

¹carlos.hvpereira@sp.senac.br

Explorando os elementos básicos

- O que sustenta a POO?



Os Pilares da POO

- **Abstração**

- Nos orienta a construir códigos que sejam relacionados com objetos da nossa vida real.

- **Encapsulamento**

- mantermos as propriedades protegidas de acessos diretamente

- **Herança**

- A Herança é uma forma de eliminar repetição de código
- um objeto pode herdar características (propriedades e métodos) de outra classe

- **Polimorfismo**

- duas ou mais classes possuem métodos com o mesmo nome, mas podendo ter implementações diferentes

Conceitos básicos de Encapsulamento

- Em vez de produzir um programa como uma única entidade **monolítica**
 - Devemos dividir o programa em várias partes **menores** e **independentes**
 - Cada parte possui implementação (são independentes entre si)
 - O encapsulamento proporciona independência:
 - Oculta detalhes internos

Formalizando Conceito

Encapsulamento:

- É a característica de **OO** de ocultar partes independentes da implementação

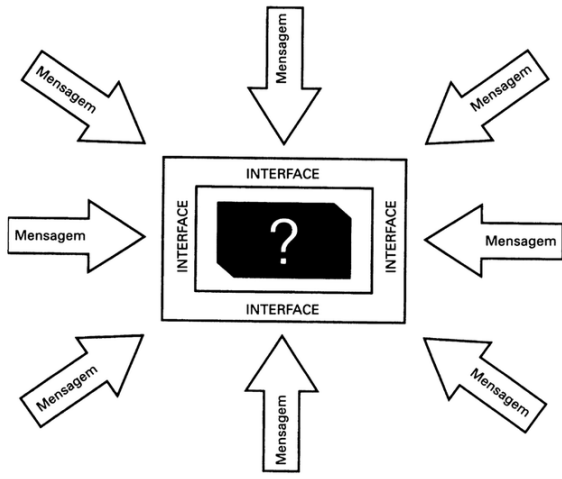


Encapsulamento na Prática

Alguns Termos

- Módulo
- Componente
- *Bean*

Encapsulamento na Prática



Encapsulamento na Prática

• Interface

- Lista os serviços fornecidos por um componente
- Tem objetivo criar um "**contrato**" onde a *Classe que a implementa* deve **obrigatoriamente obedecer**
- Define exatamente o que uma **entidade externa** pode fazer com o objeto
- **Oculta a implementação** do mundo exterior

• Implementação

- Define como um componente realmente fornece um serviço
- Define os detalhes internos do componente

Encapsulamento na Prática

Implementando *interface* em Java

- Utiliza-se palavra reservada *interface*
- Devem ser definidas em arquivos com o **mesmo nome da interface** declarada, com extensão **.java**

```
*Transportavel.java  ✕  
1  
2 public interface Transportavel {  
3     public static final String UNIDADE_VOLUME = "cm3";  
4     public String UNIDADE_PESO = "kg";  
5     //  
6     public double getVolume();  
7     public double getPeso();  
8     public int getEmpilhamentoMaximo();  
9 }
```


Encapsulamento na Prática

Implementando **Interface** em Java

- Para implementar uma interface em uma classe, utilizamos a instrução ***implements***
- Temos que implementar **todos** os métodos definidos pela interface

```
1
2 public class Mobilia implements Transportavel {
3     private String tipo;
4     private String fabricante;
5     private String material;
6     private double altura, largura, profundidade;
7     private double peso;
8     //
9     /*-----*
10    * implementação obrigatória dos métodos da interface
11    *-----*/
12    public double getVolume() {
13        double vol = this.altura * this.largura * this.profundidade;
14        return vol;
15    }
16    public double getPeso() {
17        return this.peso;
18    }
19    public int getEmpilhamentoMaximo() {
20        return 2;
21    }
22    /*-----*
23    */
```

Atividade: Encapsulamento na Prática

