Paradigma Orientado a Objeto Prof. David Tigre Moraes

Anthony Lima Campelo 061022

Isabella Alves Queiroz 060146

Iago Henrique Alves Freitas 048337

# Atividade 3

1. Quais são os quatro pilares principais da Orientação a Objeto?

* Encapsulamento: fazemos o agrupamento das coisas que fazem sentido estarem juntas, para podermos organizar e reutilizar melhor nosso código.
* Abstração: esconder os detalhes de uma implementação.
* Herança: ser uma cópia de outra classe com algumas características adicionais.
* Polimorfismo: a possibilidade de um objeto assumir diversas formas diferentes na orientação a objetos.

1. Qual a diferença entre Objeto e Classe?

Classe é uma estrutura que define características e comportamentos comuns a um grupo de objetos. Objeto é uma instância específica de uma classe.

Por exemplo, se tivermos uma classe chamada Carro, os objetos dessa classe podem representar carros específicos, como um carro vermelho ou um carro azul. Cada objeto terá características específicas (atributos), como cor, modelo e ano, e comportamentos (métodos), como ligar, desligar, acelerar etc.

1. Qual a diferença entre Override e Overload dentro da Orientação a Objeto?

Override é a prática de fornecer uma implementação específica em uma subclasse para um método que já está definido em sua superclasse. Parte do princípio de sobrescrever um método. É muito utilizado quando temos métodos estáticos em uma classe abstrata que precisam ser implementados pela classe que herda ou pela implementação de uma interface.

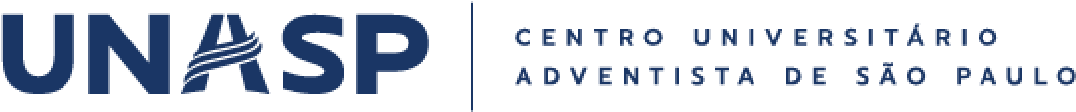
Overload é ter vários métodos com o mesmo nome dentro da mesma classe, mas com assinaturas de parâmetros diferentes. O overload é uma sobrecarga de métodos ou construtores de uma classe. Na programação orientada a objetos, esses conceitos estão ligados ao polimorfismo, pois o mesmo construtor ou método podem possuir assinaturas diferentes e retornos diferentes, no caso dos métodos.

1. O que são os métodos construtores?

Métodos construtores são métodos especiais em uma classe que são chamados automaticamente quando um objeto da classe é criado. Eles geralmente são usados para inicializar os atributos da classe. O objetivo de um método construtor é construir objetos da classe.

1. Complete a tabela abaixo com Sim e Não

|  |  |  |  |
| --- | --- | --- | --- |
|  | PUBLIC | PRIVATE | PROTECTED |
| Pode alterar/utilizar fora da classe | Sim | Não | Não |
| Pode alterar/utilizar dentro da classe | Sim | Sim | Sim |
| Os filhos podem utilizar/alterar | Sim | Não | Sim |



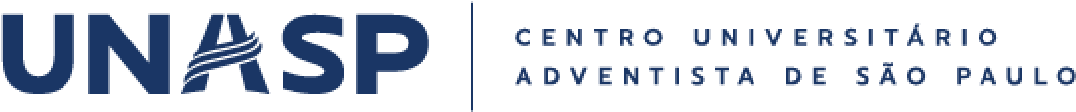
1. (FUNDATEC, 2023) O Paradigma de Orientação a Objetos (POO) é um modelo amplamente adotado em programação. Sobre essa abordagem, assinale a alternativa INCORRETA.

1. A criação de objetos é realizada através das instâncias de classes que encapsulam dados e comportamentos.
2. A abstração é uma característica importante do POO que permite a criação de entidades que representam conceitos do mundo real.
3. O conceito de herança admite que uma classe herde os atributos e métodos de outra classe, permitindo que desenvolvedores reutilizem o código e criem hierarquias de classes.
4. A associação é a relação de interação entre duas ou mais classes, na qual uma delas utiliza os serviços ou objetos da outra.
5. **Os métodos estáticos definem comportamentos ou ações que são acessados a partir dos objetos criados pela instância de uma classe.**

7. (FUNDATEC, 2023) Analise as assertivas abaixo a respeito da capacidade de sobreposição (overload) em Programação Orientada a Objetos:

1. Ela permite que vários métodos tenham o mesmo nome, mas diferentes assinaturas de parâmetros.
2. Uma subclasse pode fornecer uma implementação diferente de um método já definido em sua superclasse.
3. Está associada ao polimorfismo, permitindo que a mesma mensagem seja tratada de maneiras diferentes em classes distintas.
4. É útil quando queremos alterar o comportamento de um método em uma classe derivada, sem alterar o comportamento da classe base.

Quais estão INCORRETAS?

1. Apenas I e IV.
2. **Apenas II e III.**
3. Apenas I, II e IV.
4. Apenas II, III e IV.
5. I, II, III e IV.