

CONCEITOS DE ORIENTAÇÃO Á OBJETOS	CONCEITOS DE ORIENTAÇÃO Á OBJETOS
Quais são os <b>4 Princípios</b> que rejem a Orientação á Objetos?	<ul style="list-style-type: none"><li>- <b>Abstração</b>: o princípio que permite abstrair características e comportamentos de um objeto da vida real;</li><li>- <b>Encapsulamento</b>: o princípio que nos permite encapsular as operações que geram as abstrações, o objetivo de encapsular é manter as operações e os dados de um objeto seguros;</li><li>- <b>Herança</b>: o princípio de que á partir de uma classe modeladora podemos gerar outras classes paralelas que compartilhem dos mesmos atributos e métodos;</li><li>- <b>Polimorfismo</b>: o princípio da mudança estrutural que precisa acontecer entre as classes modeladoras e as paralelas;</li></ul>
Que <b>Ferramentas</b> a Orientação á Objetos usa?	<p>A Orientação a Objetos vai trabalhar com as seguintes ferramentas:</p> <ul style="list-style-type: none"><li>- <b>Objetos</b>: São elementos da vida real, sejam objetos físicos, comportamentos ou sentimentos, que podem ser representados por características e comportamentos;</li><li>- <b>Classes</b>: É o conjunto de características e comportamentos que modelam um objeto;</li><li>- <b>Atributos</b>: São as Características de um Objeto;</li><li>- <b>Métodos</b>: São os Comportamentos que um Objeto pode ter;</li><li>- <b>Instâncias</b>: São os objetos gerados através da modelagem das Classes;</li></ul>
Como se dá o <b>Princípio da Abstração</b> ?	<p><b>Abstração</b>: acontece quando nós visualizamos um objeto da vida real - seja um objeto físico ou abstrato - conseguimos retirar características e comportamentos dele. Por exemplo, o objeto "Motor" de um carro, que características ele teria? Poderiam ser: Peso, Tamanho, Cilindros e etc. E quanto aos comportamentos? Poderiam ser: Acelerar, Resfriar, Aquecer e etc. Tudo o que conseguirmos conceituar com características e comportamentos pode gerar Abstração. Esse é um processo essencial para darmos início a Análise Orientada á Objetos, pois, sem ter esse senso crítico não conseguiríamos gerar um projeto baseado em elementos da vida real.</p>
Como se dá o <b>Princípio do Encapsulamento</b> ?	<p><b>Encapsulamento</b>: acontece quando nós protegemos as características e os comportamentos geradores de um objeto com o objetivo de assegurar que eles não sejam danificados, podendo impactar diretamente na estrutura do objeto. Por através do Encapsulamento deixamos visível só áquilo que é relevante, segregando tudo o que não for necessário que as pessoas vejam e interajam. Por exemplo um controle remoto é formado por vários componentes como: placa eletrônica, circuitos, pilhas e etc. Mas tudo isso fica segregado, para a proteção tanto do controle, quanto do usuário, o que fica visível são os botões, ou seja, só a parte que interessa para o usuário final, isso é o encapsulamento.</p>
Como se dá o <b>Princípio da Herança</b> ?	<p><b>Herança</b>: acontece quando temos uma Classe com características que podem ser herdadas por outras Classes, usando o método de Classe Mãe e Classe Filha, onde a Classe Mãe é uma Generalização e as Classes Filhas são Especializações da Classe Mãe. Esse princípio preza pelo reaproveitamento de código e boa identificação entre as Entidades de um projeto. Onde o código de uma Classe Mãe pode ser reaproveitado pelas Classes Filhas, e a identificação de Entidades paralelas possa ser facilmente referenciada por através das heranças.</p>
Ilustre o Princípio da Herança	<pre>graph BT; Pessoa[Pessoa Class] --&gt; Professor[Professor Class]; Pessoa --&gt; Aluno[Aluno Class];</pre> <p>O diagrama ilustra o princípio da herança com a seguinte estrutura:</p> <ul style="list-style-type: none"><li><b>Pessoa Class</b> (Super Classe) com propriedades: Idade, Nome, RG.</li><li><b>Professor Class</b> (Sub classes) herda de Pessoa, com propriedades: Disciplinas, Matricula, e métodos: Lançar Falta, Lançar Nota.</li><li><b>Aluno Class</b> (Sub classes) herda de Pessoa, com propriedades: RA, e métodos: Consultar Falta, Consultar Nota.</li></ul> <p>As setas indicam a direção da herança: Generalização (para cima) e Especialização (para baixo).</p>
- reservada para a questão acima -	
Como se dá o <b>Princípio do Polimorfismo</b> ?	<p><b>Polimorfismo</b>: acontece quando nós modificamos um método de uma classe para que ele assuma outro comportamento dentro de outra classe - e as vezes dentro da própria classe caso sejam um polimorfismo de sobrecarga - sem alterar o nome do método e até sem alterar a assinatura do método. Esse princípio preza por manter o nome de um método mesmo que ele tenha comportamentos diferentes para cada caso em cada classe especializada.</p>
Ilustre o Princípio do Polimorfismo	<pre>graph BT; FiguraGeometrica[Figura Geométrica Class] --&gt; Quadrado[Quadrado Class]; FiguraGeometrica --&gt; Triangulo[Triangulo Class];</pre> <p>O diagrama ilustra o princípio do polimorfismo com a seguinte estrutura:</p> <ul style="list-style-type: none"><li><b>Figura Geométrica Class</b> (Super Classe) com propriedades: Quantidade Lados, e método: Desenha.</li><li><b>Quadrado Class</b> (Sub classes) herda de Figura Geométrica, com método: Desenha.</li><li><b>Triangulo Class</b> (Sub classes) herda de Figura Geométrica, com método: Desenha.</li></ul>
- reservada para a questão acima -	