Paradigma de java orientada a objectos

<<Lucinda Rodrigues / Nº 1000021599>>  
Faculdade de Engenharia   
Universidade Católica de Angola  
<<lucindalemos3@gmail.com>>

*Abstrato*— O presente trabalho refere-se aos conceitos que o java orientado a objectos utiliza para poder manipular melhor os atributos ,classes , métodos , dos seus objectos e explica como devem ser ulilizados .

*PalavrasChaves<<Classes,Objectos,Atributos,Métodos>>*

# Introdução

Actualmente o paradigma de java orientada a objectos é o modelo de programação mais adotado no desenvolvimento de sistemas corporativos ,ele é utilizado com objectivo de obter-se alguns benefícios específicos ,tal como facilitar a manutenção das aplicações. Ele engloba alguns conceitos para que ajudam a alcançar estes objectivos , tais como: **Encapsulamento, Abstração, Polimorfismo, Herança, Interface.**

# Conceitos

Encapsulamento : É um conceito da Programação Orientada a Objetos onde o estado de objetos (as variáveis da classe) e seus comportamentos (os métodos da classe) são agrupados em conjuntos segundo o seu grau de relação, ou seja serve para controlar o acesso aos atributos e métodos de uma classe.

Herança: É permitido que uma classe herde atributos e métodos da outra. Os modificadores de acessos das classes, métodos e atributos só podem estar com visibilidade public e protected para que sejam herdados. Uma das grandes vantagens de usar o recurso da herança é na reutilização do código. Esse reaproveitamento pode ser acionado quando se identifica que o atributo ou método de uma classe será igual para as outras. Para efetuar uma herança de uma classe é utilizada a palavra reservada chamada extends.

Polimorfismo: significa "muitas formas", é o termo definido em linguagens orientadas a objeto e essas formas, em nosso contexto de programação, são as subclasses/objetos criados a partir de uma classe maior, mais geral ou abstrata. Ou seja é a capacidade que o Java nos dá de controlar todas as formas de uma maneira mais simples e geral, sem ter que se preocupar com cada objeto especificamente.

Interface: É o contacto entre a classe e o mundo exterior , ou seja é um bloco de código que define um tipo , métodos e atributos que este tipo deve possuir .A interface define um padrão para especificação do comportamento de classes , porém os métodos de uma interface são implementados de forma particular em cada classe.

Abstração: É utilizada para definir entidades do mundo real, ou seja ao criarmos um objecto , esse deve ter uma entidade única dentro do sistema para evitar conflitos , também deve possuir propriedades que o definem e por fim métodos que são as ações que ele ira realizar

##### Referências Bibliográficas

1. <https://www.devmedia.com.br/os-4-pilares-da-programacao-orientada-a-objetos/9264>
2. <https://medium.com/@TDamiao/16-conceitos-poo-programa%C3%A7%C3%A3o-orientada-a-objeto-6cdc72ac3ee2>
3. [www.k19.com.br](http://www.k19.com.br)
4. <https://pt.wikipedia.org/wiki/Abstra%C3%A7%C3%A3o_(ci%C3%AAncia_da_computa%C3%A7%C3%A3o)>