

Para enfrentar problemas de estrutura surgiu a programação orientada a objetos.

Onde representa os sistemas em modelo de objetos integrantes, melhorando o software de maneira geral. Dentro de Orientação a objetos temos diversos conceitos, como:

- Objeto:

Representam instâncias de classes, com características e formas de manipulação, representam uma entidade do mundo real, como um livro por exemplo.

- Atributos e Métodos:

São os valores e propriedades de um objeto, os atributos são as características como nome, idade, peso, etc. Já os métodos são as ações que tal objeto pode exercer, como andar, falar, dormir.

- Classes:

São os agrupamentos de objetos, quando um ou mais objetos possuem as mesmas características, podemos dizer que fazem parte da mesma classe. Podemos incluir também o conceito de hierarquia de classes, onde uma superclasse passa atributos para as subclasses.

- Instanciação:

É quando produzimos objetos, utilizando como base uma classe existente, como uma espécie de gabarito.

- Classes puras ou abstratas:

São as classes onde objetos não são instanciados diretamente, ela apenas serve como base para estruturação.

- Notação gráfica das classes e dos objetos:

É o padrão para a declaração do diagrama, seguindo sempre o mesmo método de notação.

Somos introduzidos a Java, uma linguagem para desenvolvimento de software que funciona em conjunto com a internet. Junto a essa explicação temos as formas de como são definidas as classes, e a definição de cada componente ali presente na estrutura como, public, static, void.

Temos como princípios básicos da orientação a objetos:

- Abstração:

É uma forma de ocultar informação, tornando público somente aquilo necessário para a operação, isolamos o objeto e passamos somente o que é relevante para a situação.

- Encapsulamento:

É uma forma de proteger os dados, e desacoplar as funções deixando separado em cada método o que aquele objeto faz, não é necessário alterar o objeto diretamente, somente utilizar dos métodos já dispostos ou implementar novos.

- Herança:

É o conceito de herdar as características de um para outro, na programação orientada a objetos, temos superclasses que passam os atributos para as subclasses que herdam para si, os atributos da superclasse. Dentro de herança temos diversos tipos de herança como:

Herança simples:

Onde herda-se os atributos somente de uma superclasse, como exemplo, funcionário herda de pessoa, possuindo assim os mesmos atributos de pessoa e os seus próprios a diferenciando das demais classes.

Herança múltipla:

Pode-se herdar de mais de uma superclasse, sendo preciso criar outra classe com o nome das superclasses que foi herdada, ele recebe os atributos das duas superclasses e pode ter os seus próprios também.

- Polimorfismo:

Temos objetos de diversas classes que possuem métodos em comum mas que são executados de forma diferente, onde os objetos respondem a uma mesma mensagem de forma diferente.

- Estrutura:

É uma forma de ajudar na organização do problema, tendo duas formas de estrutura, Generalização e Todo-Parte.

Estrutura de Generalização - Especialização:

Formada por uma classe genérica no topo e suas subclasses especializadas dispostas abaixo. Utilizando como exemplo a superclasse pessoa, fica ao topo e abaixo as subclasses Funcionário e Cliente, que são especializações de Pessoa.

Estrutura Todo – Parte:

É feita a decomposição do objeto, onde seus componentes podem se tornar outros objetos para um melhor entendimento do problema em questão.