



Orientação a Objetos

Stallos Tecnologia



O que é orientação a Objetos ?

- “Uma maneira de modelar o mundo real.”
- “Uma forma de ter organização do código”
- “Uma forma de poder representar todo o mundo real por meio de objetos”



Programação Estruturada

- Paradigma que permitia que todo programa fosse basicamente desmembrado em três ações básicas.
 - Sequência: Instruções executadas uma por vez em que aparecem no programa
 - Decisão: Instruções condicionais que introduzem lógica ao programa.
 - Iteração: Instruções executadas várias vezes, conhecidas como loopings.
- Exemplos de linguagens (C, Basic, Pascal).
- Vantagens do Modelo
 - Os problemas a serem resolvidos podem ser representados em uma única estrutura.
- Desvantagens do Modelo
 - Os dados são separados da estrutura do programa quando alterado necessita-se refatorar todo ou maior parte do código.



Programação Orientada a Objetos

- Orientado ao ambiente real propõe a decomposição em pequenas unidades de software baseado no ambiente, denominando essas unidades de objetos.
- Linguagens (C++, C#, Java, Object Pascal, Python)
- Vantagens do Modelo
 - Alterações nas unidades de código não devem causar impacto a nível de refatoração de todo o sistema.
- Desvantagens do Modelo
 - Aumenta a complexidade na visualização do software, necessitando documentações complementares para ser mantido.



Objeto e Classes

- O que é uma classe ?
 - Uma representação das entidades que interagem no ambiente de forma descritiva.
- O que é um Objeto ?
 - Podemos definir um objeto como a materialização, o resultado final de uma classe depois de criado e armazenado em memória.



Pilares da Orientação a Objetos

Os pilares da orientação a objetos são quatro sendo eles a base essencial para a implementação de um software escrito seguindo o paradigma de orientação a objetos.

- Abstração.
- Encapsulamento.
- Herança (Generalização).
- Polimorfismo.



Abstração

- Abstração é considerada a primeira fase da análise orientada a objetos , consiste em concentrar a observação em aspectos essenciais do ambiente para incorporar no ambiente orientado a objetos somente o necessário.
- Métodos para esta etapa
 - Diagrama de caso e uso
 - Papel e caneta
- Exemplo 1 - Imagine um ambiente em que controlamos livros de uma biblioteca apesar dos mesmos terem (Autor, Editora, Nome, Número de Páginas, Cor da Capa, etc..), para nosso ambiente necessitamos somente de Autor, Editora e Nome.



Encapsulamento

- Encapsulamento corresponde ao nível de segurança da sua classe quando a seu comportamento, expondo somente o que as entidades externas precisam saber.
 - Como exemplo podemos citar uma casa pelas suas janelas podemos ver os objetos dentro da mesma porém, vemos apenas os objetos exibidos pela janela.
- Exemplo 2 - Imagine um ambiente em que devemos controlar a maioridade do usuário e devemos saber em pontos do sistema se o usuário é maior de idade para adotar algumas ações.



Herança

- A herança tem por finalidade fornecer um mecanismo em que seja possível o reaproveitamento do código que corresponde a uma carga hereditária.
- A herança é constituída por superclasses (pai) e subclasses (filho), o pai dita a carga hereditária para o filho, mas o filho não tem poder sobre o pai.
- Exemplo 3 - Como exemplo podemos adotar um ambiente que lide com veículos automotores, ambos teriam propriedades semelhantes agrupadas em um pai Veículo porém podem ser especializadas em (Carros, Motos, etc).



Polimorfismo

- Polimorfismo é um dos pilares mais importantes da orientação a objeto, permitindo o comportamento de várias formas de um objeto, desde-que especializado.
- Polimorfismo só é possível por meio dos mecanismos de reescrita, sobrecarga e herança
 - Reescrita : Tem por finalidade reescrever o comportamento do objeto
 - Sobrecarga: Tem por finalidade adicionar mais métodos a classe.
- Exemplo 4 - Como este exemplo podemos adotar o exemplo anterior, alterando o comportamento dos métodos utilizando reescrita e sobrecarga.



Contratos e Herança Múltipla

- Algumas Linguagens orientadas a objetos permitem a criação de contratos, estes impõem a estrutura do objeto, porém não ditam seus comportamentos.
 - Em C# chamamos estes contratos de Interfaces estas interfaces, ditam a estrutura da classes obrigando a implementação das propriedades e métodos nela descritos
- Algumas linguagens permitem herança múltipla de suas classes, entretanto em C# este comportamento só é possível por meio de contratos (Interfaces).
- Utilizado para casos em que uma entidade pode ser representada por mais de uma forma
- Exemplo 5 - Para este exemplo podemos considerar uma máquina que efetue movimentos em quatro direções (Frente, Trás, Esquerda e Direita).



O que é orientação a Objetos ?

“A Orientação a objetos é a habilidade de obter controle absoluto, através do uso do polimorfismo, sobre cada dependência de código fonte do sistema.”

Martin, Robert C.. Arquitetura Limpa (Robert C. Martin) . Alta Books. Edição do Kindle.