

# Metodologias Orientadas a Objetos (parte 2)



Um processo de desenvolvimento de software não começa escrevendo-se o código de um programa. Na verdade, um processo sistemático de análise e planejamento vem primeiro. A programação é apenas uma parte do processo intelectual associado à construção de software e à ciência da computação.



Através de uma abordagem orientada por objetos, você será apresentado à noção simples e atrativa de que um sistema de software é composto de objetos interagindo de maneira harmoniosa que se comunicam entre si através de mensagens. Essas mensagens são definidas com precisão numa descrição de uma classe.



A orientação a objetos é um paradigma de análise, projeto e programação de sistemas de software baseado na composição e interação entre diversas unidades de software chamadas de objetos.



o paradigma "orientação a objeto", tem bases conceituais e origem no campo de estudo da cognição, que influenciou a área de inteligência artificial e da linguística, no campo da abstração de conceitos do mundo real



Melhor estratégia de modelagem para se eliminar o "gap semântico", dificuldade recorrente no processo de modelar o mundo real do domínio do problema em um conjunto de componentes de software que seja o mais fiel na sua representação deste domínio



Facilitaria a comunicação do profissional modelador e do usuário da área alvo, na medida em que a correlação da simbologia e conceitos abstratos do mundo real e da ferramenta de modelagem (conceitos, terminologia, símbolos, grafismo e estratégias) fosse a mais óbvia, natural e exata possível



Na programação orientada a objetos, implementa-se um conjunto de classes que definem os objetos presentes no sistema de software. Cada classe determina o comportamento (definido nos métodos) e estados possíveis (atributos) de seus objetos, assim como o relacionamento com outros objetos.



#### Linguagens 00

C++, C#, VB.NET, Java, Object Pascal, Objective-C, Python, SuperCollider, Ruby e Smalltalk



#### Suporte a 00

ActionScript, ColdFusion, Javascript, PHP (a partir da versão 4.0), Perl (a partir da versão 5), Visual Basic (a partir da versão 4) e Delphi (a partir da versão 6)



#### Conceitos essenciais

Classe representa um conjunto de objetos com características afins. Uma classe define o comportamento dos objetos através de seus métodos, e quais estados ele é capaz de manter através de seus atributos. Exemplo de classe: Os seres humanos.



#### Subclasse

Subclasse é uma nova classe que herda características de sua(s) classe(s) ancestral(is)



Uma classe para Orientação à Objetos (O.O) é como uma receita de bolo, você não come uma receita de bolo, mas a usa como referência para preparar o bolo. A função da classe é semelhante a receita do bolo a utilizamos para criar um objeto bolo a partir de suas especificações.



Objeto / instância de uma classe. Um objeto é capaz de armazenar estados através de seus atributos e reagir a mensagens enviadas a ele, assim como se relacionar e enviar mensagens a outros objetos. Exemplo de objetos da classe Humanos: João, José, Maria



Atributo são características de um objeto. Basicamente a estrutura de dados que vai representar a classe. Exemplos:

Funcionário: nome, endereço, telefone, CPF,...;

Carro: nome, marca, ano, cor, ...;

Livro: autor, editora, ano.



os atributos possuem valores. Por exemplo, o atributo cor pode conter o valor azul. O conjunto de valores dos atributos de um determinado objeto é chamado de estado



Método definem as habilidades dos objetos. Totó é uma instância da classe Cachorro, portanto tem habilidade para latir, implementada através do método deUmLatido.



o ato da criação do objeto pela classe é conhecido como "instanciar o objeto". Podemos criar centenas de bolos a partir da mesma receita (classe), eles podem ser bem semelhantes, alguns idênticos, mas são objetos diferentes.



#### Coisas Legais:

Livro C# para Crianças

http://msdn.microsoft.com/pt-br/beginner/bb308756(printer).aspx



#### http://jedi.wv.com.br/login/signup.php?

- Introdução a Programação I
- Introdução a Programação II
  - Estruturas de Dados



Um método em uma classe é apenas uma definição. A ação só ocorre quando o método é invocado através do objeto, no caso Totó



Dentro do programa, a utilização de um método deve afetar apenas um objeto em particular; Todos os cachorros podem latir, mas você quer que apenas Bidu dê o latido. Normalmente, uma classe possui diversos métodos, que no caso da classe Cachorro poderiam ser sente, coma, deite, role, pegue...



Mensagem é uma chamada a um objeto para invocar um de seus métodos, ativando um comportamento descrito por sua classe. Também pode ser direcionada diretamente a uma classe (através de uma invocação a um método estático)



Herança (ou generalização) é o mecanismo pelo qual uma classe (sub-classe) pode estender outra classe (super-classe), aproveitando seus comportamentos (métodos) e variáveis possíveis (atributos). Um exemplo de herança: Mamífero é superclasse de Humano. Ou seja, um Humano é um mamífero. Há herança múltipla quando uma sub-classe possui mais de uma superclasse. Essa relação é normalmente chamada de relação "é um"



Associação é o mecanismo pelo qual um objeto utiliza os recursos de outro.
Pode tratar-se de uma associação simples "usa um" ou de um acoplamento "parte de". Por exemplo:
Um humano usa um telefone. A tecla "1" é parte de um telefone



Encapsulamento consiste na separação de aspectos internos e externos de um objeto. Este mecanismo é utilizado amplamente para impedir o acesso direto ao estado de um objeto (seus atributos), disponibilizando externamente apenas os métodos que alteram estes estados. Exemplo: você não precisa conhecer os detalhes dos circuitos de um telefone para utilizá-lo. A carcaça do telefone encapsula esses detalhes, provendo a você uma interface mais amigável (os botões, o monofone e os sinais de tom)



Abstração é a habilidade de concentrar nos aspectos essenciais de um contexto qualquer, ignorando características menos importantes ou acidentais. Em modelagem orientada a objetos, uma classe é uma abstração de entidades existentes no domínio do sistema de software



Polimorfismo consiste em quatro propriedades que a linguagem pode ter (atente para o fato de que nem toda linguagem orientada a objeto tem implementado todos os tipos de polimorfismo):



#### **Universal:**

Inclusão: um ponteiro para classe mãe pode apontar para uma instância de uma classe filha (exemplo em Java: "List lista = new LinkedList();" (tipo de polimorfismo mais básico que existe)

Paramétrico: se restringe ao uso de templates (C++, por exemplo) e generics (Java/C#)

#### Ad-Hoc:

Sobrecarga: duas funções/métodos com o mesmo nome mas assinaturas diferentes

Coerção: a linguagem que faz as conversões implicitamente (como por exemplo atribuir um int a um float em C++, isto é aceito mesmo sendo tipos diferentes pois a conversão é feita implicitamente)



Interface é um contrato entre a classe e o mundo externo. Quando uma classe implementa uma interface, ela está comprometida a fornecer o comportamento publicado pela interface



## Pacotes (ou Namespaces) são referências para organização lógica de classes e interfaces



#### Ou seja...

### Objeto, objetos em todos os lugares Objetos ordinários

O que é um objeto? Um objeto ordinário é caracterizado tanto por seu comportamento quanto pelo seu estado interno. Para objetos ordinários, há uma linha que separa o interior do objeto do lado de fora deste. Dentre as características que definem objetos ordinários estão incluídas textura, cor, cheiro, som, ou custo.



Uma criança no início de sua vida é capaz de distinguir objetos que estão ao seu redor. Isto inclui seus pais e as pessoas que tomam conta dela e os objetos próximos ou sobre seu berço. Isto quer dizer que os seres humanos são criaturas orientadas por objetos.



À medida que nós envelhecemos nós aprendemos também a caracterizar objetos baseando no seu aspecto e comportamento. Nossa orientação por objetos é fortalecida pela a observação do mundo a nossa volta e, mais tarde, por um processo formal de educação.



Por exemplo, em Química nós aprendemos, a compreender um átomo em particular baseados em sua classificação numa Tabela Periódica de Elementos. Em Biologia, nós aprendemos, a classificar as várias espécies de organismos vivos baseados em uma elaborada classificação hierárquica de espécies.



Quando jovens, nós observamos os objetos a nossa volta, nós frequentemente distinguimos ou classificamos objetos por suas semelhanças e diferenças de aspecto e comportamento. Mas, surpreendentemente, nós rapidamente aprendemos a classificar tipos similares de objetos como um carro independente de seu aspecto, tamanho, cor ou textura precisos.



Uma criança pode naturalmente reconhecer um desenho mal feito de um carro no livro, um carrinho de brinquedo do tamanho de uma caixinha de fósforos, e um carro real bem como sendo diferentes exemplos, encarnações, ou "instâncias" da classificação CARRO.



Muitas crianças podem distinguir mais tarde um carro de um caminhão baseando- se não apenas em sua forma precisa, tamanho, cor, ou outro dado "significativo" de um caminhão, mas ao invés, baseando-se em algumas das propriedades visuais abstratas de carros e caminhões. Parece uma característica dos seres humanos a habilidade natural de classificar objetos. Nós chamaremos estas classificações de classes.



A palavra instância será usada para significar um membro particular de uma classe. Por exemplo, um carro vermelho do tamanho de uma caixa de fósforo que seja o brinquedo favorito de uma criança é uma instância da classe CARRO (serãd usadas letras maiúsculas para classes) da mesma maneira que o esboço mal feito de um carro no livro de uma criança ou o automóvel real na garagem são outras instâncias de CARRO.



Muitas crianças desenvolvem um nível alto de entusiasmo para carros de brinquedo, caminhões, aviões, barcos, e trens. Esta fascinação parece estar baseada na habilidade comum de todos estes objetos moverem-se de um local a outro e, a mais importante, a habilidade que a criança tem de controlar este movimento. Cedo, as crianças são hábeis em obter uma entendimento abstrato de um VEÍCULO.



Esta abstração representa os traços comuns a todos os veículos incluindo carros, caminhões, aviões, barcos e trens (e outros tipos de veículos conhecidos mais tarde, no momento em que sua experiência de vida e poder aquisitivo aumentam).



Uma criança parece apta para fazer esta generalização antes de aprender a palavra "veículo". Como humanos parecemos ser capazes de desempenhar essa tais generalizações de maneira natural. Esta abordagem de classificação serve como base para uma abordagem "orienta por objetos" para solução de problemas.



A classe VEÍCULO, seja formalmente definida ou entendida informalmente, é considerada como sendo uma classe abstrata em contraste com uma classe concreta, tal como CARRO, CAMINHÃO, AVIÃO, BARCO, E TREM



Uma classe abstrata (como veículo) é aquela que não possui instâncias de verdade, mas pode ser usada para produzir instâncias de classes concretas (como CARRO, CAMINHÃO, AVIÃO, BARCO, ou TREM). Ela é uma generalização de uma classe concreta.



A abstração da classe VEÍCULO contêm características comuns de CARRO, CAMINHÃO, AVIÃO, BARCO e TREM, formando a base para a noção de herança. A classe concreta adquire (herda) características de sua classe pai (a classe abstrata). Cada característica herdada da classe abstrata é achada em instâncias na classe concreta.



Em nossa percepção dos objetos ordinários, é difícil ou quase impossível, definir precisamente as características de cada classe concreta. Nós podemos rapidamente desenvolver a habilidade de identificar um carro com exatidão. Nós podemos nunca desenvolver a habilidade de definir a classe CARRO com precisão. Nossas habilidades de reconhecimento de padrões são baseadas em fenômenos complexos que não podem ser modelados facilmente. Quando nós "modelamos" uma classe CARRO, nós tentamos extrair características essenciais, ignorando os detalhes não essenciais. Entretanto, é confortante saber que embora não possamos modelar um carro com precisão, podemos ao menos guiá- lo.



Quando começamos a pensar, escrever ou falar sobre carros, nós desenvolvemos uma abstração desta entidade. Esta não é uma atividade que vem facilmente ou cedo na vida. Tanto artistas quanto engenheiros precisam desenvolver tal abstração quando tentam representar um carro. Cada um fará isto de maneira diferente. O artista irá enfatizar o contorno, a textura e a cor ao passo que o engenheiro irá enfatizar o formato e o comportamento do carro. O engenheiro, em particular, se preocupa com a relação entre formato e comportamento. O modelo abstrato de um carro desenvolvido por um engenheiro deve unificar formato e comportamento. Esta unificação é chamada de encapsulamento.



Os aspectos referentes à forma, textura e cor devem ser considerados como os "dados" do carro. Além desses "dados", o engenheiro se preocupa com questões como a capacidade de fazer curvas do carro, sua capacidade de frear, sua aceleração, etc. Estes fenômenos envolvem a reação do veículo a vários estímulos (por exemplo, pisar no pedal do acelerador, pisar no pedal do freio, virar o volante, etc.). O comportamento de um carro é fortemente influenciado por esses "dados". Um carro grande e pesado geralmente requer muito mais potência para ter uma determinada capacidade de aceleração que um carro pequeno e leve. Ele é geralmente menos manobrável que um carro leve.



Para entender as características de aceleração de um carro, os únicos "dados" (daqui em diante chamados de atributos) que podem ser relevantes são a massa do carro, torque, fricção dos pneus e coeficiente aerodinâmico. Essas variáveis constituem o estado interno ou atributos da classe CARRO. Outros atributos como sua cor, marca ou preço são irrelevantes. Aspectos como a velocidade do carro podem ser calculados através do conhecimento desses atributos.



Se alguém está desenvolvendo uma abstração onde o carro é um produto comercial, então os atributos que nós devemos usar para representar o estado interno do carro incluem o valor de empréstimo, a taxa de juros do empréstimo, o número de meses do empréstimo o número de pagamentos já feitos, e o preço de tabela do carro. Por esses atributos, o "comportamento" do carro como produto comercial pode ser totalmente descrito.



A modelagem de objetos é similar à modelagem de quaisquer entidades da ciência. O nível de detalhes definidos no modelo depende das metas do problema Se é desejado estudar as propriedades termodinâmicas da combustão de 4 tempos, associada a um motor de combustão interna, então um modelo que inclui os mínimos detalhes é apropriado. Isso incluiria informações sobre a geometria de cada cilindro e a geometria de cada pistão.



Portanto, a descrição de um objeto, a abstração do objeto, é baseada no problema onde o objeto existe. Aspectos do objeto que exercem um papel essencial na descoberta de uma solução para o problema dado devem ser representados no modelo de objeto (a classe) e os aspectos que não são essenciais são ignorados. Uma abstração representa uma descrição simplificada da realidade. O Dicionário Oxford (1966) sugere o seguinte sobre abstrações: "O princípio de ignorar aqueles aspectos de um assunto que não são relevantes para o problema a fim de se concentrar mais naqueles que são."



Leiam o texto "livro normal"