

# Estruturas e relacionamentos

- ✓ Generalização e herança;
- ✓ Agregação;
- ✓ Ligações e associações;
- ✓ Compartilhamento.

### 4.1. Introdução

Nesta leitura complementar, apresentaremos uma parte do Modelo de Objetos (um subconjunto do modelo OMT), que retrata o relacionamento entre objetos. Esse modelo pode ser introduzido a partir do estabelecimento dos principais conceitos e definições da abordagem de orientação a objetos.

### 4.2. Generalização e herança

A generalização e a herança são abstrações que permitem que classes compartilhem similaridades ao mesmo tempo em que preservam as características que as diferem.

O relacionamento de uma classe com suas versões refinadas, ou seja, especializadas, é chamado de generalização. Nesse ponto, é importante definir dois conceitos:

- **Superclasse ou classe base:** É a classe por meio da qual podemos gerar subclasses; também é conhecida como classe pai;
- **Subclasse ou classe derivada:** É a versão refinada de uma superclasse.

Vejamos um exemplo: pode haver uma classe base chamada **Pessoa**, com características comuns a todas as pessoas: **Nome**, **Email**, **Telefone**. Uma classe derivada poderia ser a classe **Funcionario**, que, além de ter os campos **Nome**, **Email** e **Telefone**, teria também os campos **Salário** e **Cargo**. Uma outra classe derivada poderia ser a **Cliente**, que, além dos campos herdados da classe base, teria o campo **Nome do Contato** ou **Celular**.



As classes base melhoram a consistência de um sistema, facilitando a padronização dos métodos e propriedades. Os principais ambientes de programação do mercado, como o .NET Framework da Microsoft ou o Java da Sun Systems, foram construídos usando o sistema de classes base e classes derivadas.

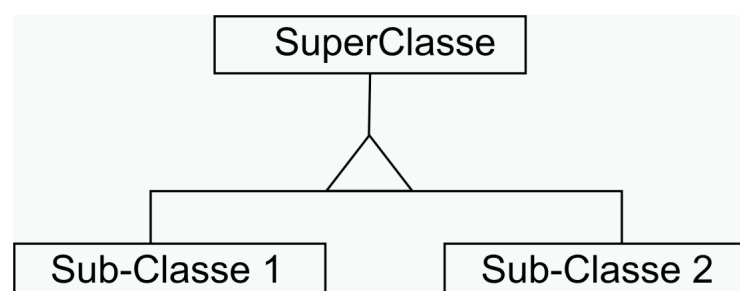
A generalização também pode ser chamada de relacionamento **is-a** (ou seja, **é-um**), já que toda instância de classe derivada também é uma instância de classe base. As características da classe base são herdadas pela classe derivada, e as operações e os atributos comuns a um conjunto de classes derivadas são colocados como atributos e operações da classe base.



ILUSTRAÇÃO: EDUARDO JOSÉ DE SOUZA ENGELMANN

A generalização é a estrutura que permite a utilização do conceito de herança, sendo aplicada a diversos níveis. Além das características herdadas de seus ancestrais, uma classe derivada pode acrescentar operações e atributos específicos.

A imagem seguinte exibe a notação diagramática de OMT utilizada para representar a generalização: um triângulo cujo vértice aponta para a classe base.

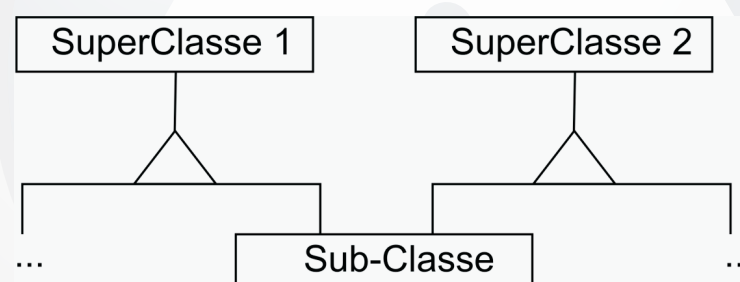


Cada associação do tipo generalização pode possuir um discriminador associado. O discriminador é um atributo (do tipo enumeração) utilizado para indicar qual é a propriedade do objeto que é abstraída pelo relacionamento de generalização. Trata-se de um nome para a base de generalização.

Uma característica de classe base pode ser sobreposta por uma classe derivada, caso esta defina uma característica própria com o mesmo nome. A característica própria da classe derivada refinará e substituirá a característica da classe base.

As características podem ser sobrepostas por questões de desempenho ou de refinamento de especificação, por exemplo. As características que podem ser sobrepostas são os valores default dos atributos e métodos de operação, entre outras. É importante lembrar que uma característica não deve ser sobreposta de forma inconsistente com a semântica da classe base.

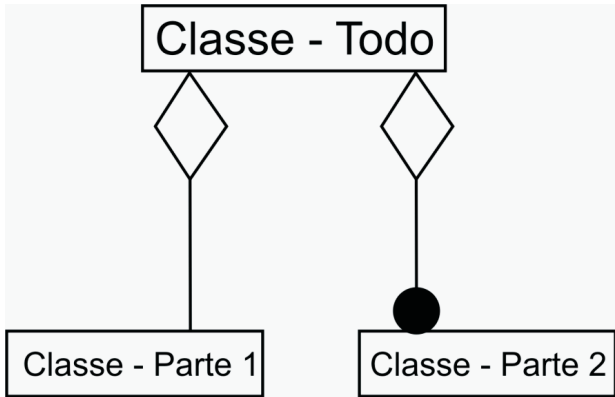
Temos a herança múltipla representada graficamente na imagem a seguir:



A herança múltipla não é aceita em todos os ambientes de programação orientados a objetos.

### 4.3. Agregação

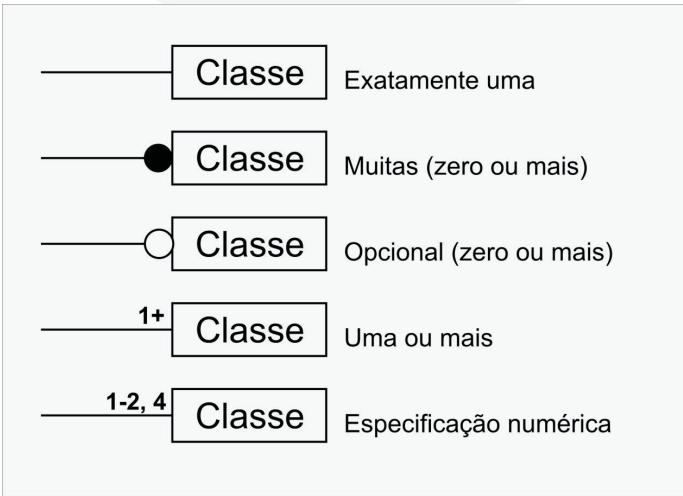
Na agregação, há certa coesão entre as partes, porém, elas não são totalmente dependentes. Para exemplificar, podemos imaginar a criação de uma nova classe, **Família**, constituída de vários membros da classe **Pessoa**, porém, os membros da classe **Pessoa** podem existir fora da classe **Família**.



A agregação define um relacionamento do tipo **uma-parte-de**. Nesse tipo de relacionamento, alguns objetos (parte) representam componentes de outro objeto (todo), porém, um não contém o outro, o todo não contém a parte, não havendo relação de posse exclusiva. Em uma agregação, um componente que faz parte de outro pode existir isoladamente.

Isso é diferente do que acontece na composição, como em uma nota fiscal, por exemplo, em que o objeto que representa um produto comprado pertence a essa nota e apenas a ela, não fazendo sentido esse objeto existir se não for dentro dessa nota fiscal.

As representações gráficas para tipos diferentes de agregação são exibidas na próxima imagem. Essas notações também podem ser utilizadas para associações e para ligações.



### 4.3.1. Conexões entre objetos

Conexões de ocorrência e conexões de mensagens são tipos de conexão entre objetos que não caracterizam um tipo de hierarquia ou de estrutura.

#### 4.3.1.1. Conexão de ocorrência

Este tipo de conexão ocorre quando um atributo de um objeto faz referência a outro objeto. As conexões de ocorrência normalmente são criadas quando identificamos que atributos redundantes de um objeto são parte de outro objeto.

As conexões de ocorrência possibilitam também o registro do número de vezes que um objeto faz referência a outro ou é referenciado, o que também é chamado de cardinalidade.

#### 4.3.1.2. Conexão de mensagem



ILUSTRAÇÃO: EDUARDO JOSÉ DE SOUZA ENGELMANN

Este tipo de conexão ocorre quando um objeto envia uma mensagem para outro objeto. No contexto de uma conexão de mensagem, existem os seguintes elementos:

- **Objeto transmissor:** O objeto que transmite a mensagem para um outro objeto;
- **Mensagem:** Procedimento que dispara um método específico para que o objeto receptor execute um comportamento determinado. Pode ser uma solicitação de informações ou uma solicitação para a realização de alguma ação no objeto;
- **Objeto receptor:** O objeto que recebe a mensagem e retorna ou não uma resposta para o objeto transmissor.



## 4.4. Ligações e associações

Ligações e associações são mecanismos que estabelecem relacionamentos entre objetos e classes. A seguir, cada um desses mecanismos será descrito:

- **Ligações**

Uma ligação conecta duas instâncias de objetos, de forma física ou conceitual, por exemplo, **Paulo** é aluno da **Faculdade\_Impacta**. Uma ligação é uma instância de uma associação.

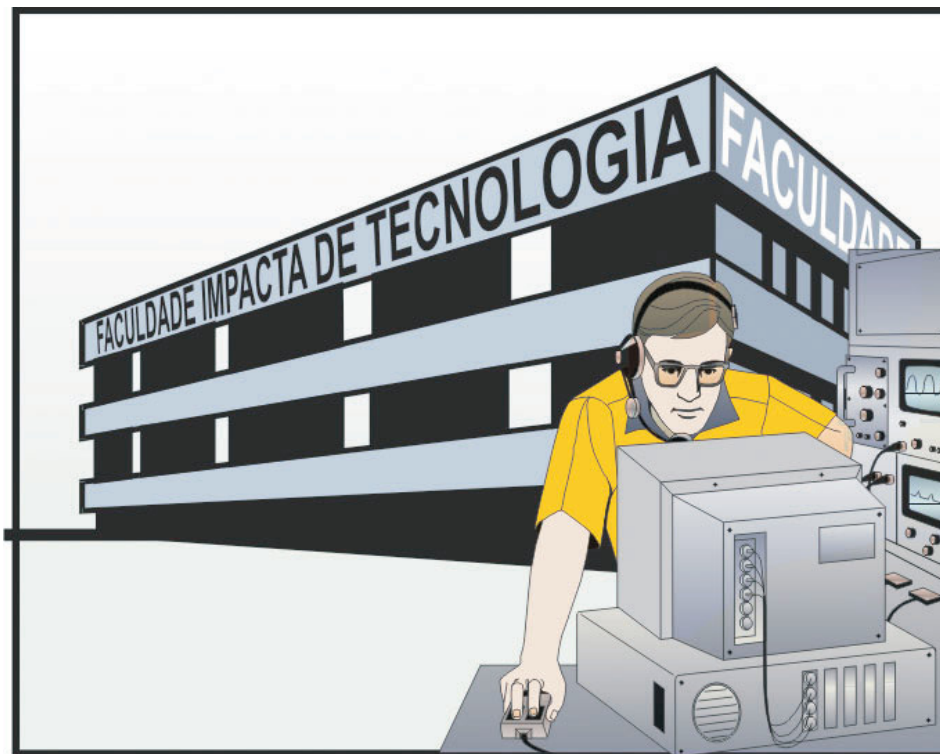
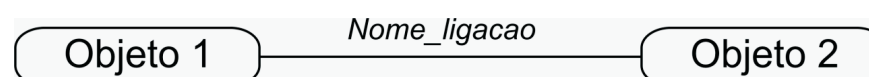


ILUSTRAÇÃO: EDUARDO JOSÉ DE SOUZA ENGELMANN

Na notação de diagramas OMT, uma ligação é representada como uma linha que conecta dois objetos. A imagem seguinte exemplifica um diagrama OMT com ligação:



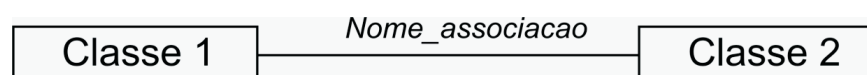
- **Associações**

As associações definem um conjunto de ligações que compartilham a mesma semântica e estrutura, por exemplo, **uma pessoa é aluna de uma faculdade**. Podemos entender que, assim como as classes descrevem grupos de objetos potenciais, as associações descrevem grupos de ligações potenciais.

# Introdução à Programação Orientada a Objeto (online)

66

Uma associação é representada por uma linha que conecta duas classes. A próxima imagem exemplifica um diagrama OMT com associação:



Na notação de diagramas **OMT**, normalmente, os nomes das associações e das ligações são exibidos em estilo itálico e podem ser omitidos quando só existir uma associação de sentido óbvio entre um par de classes, para as associações, ou entre um par de objetos, para as ligações.

A OMT introduz o conceito de atributo de ligação para os casos em que os atributos dizem respeito a associações e não a classes. A associação pode ser modelada como uma classe conectada à associação quando possuir operações associadas:

