



Programação Orientada a Objetos em C#

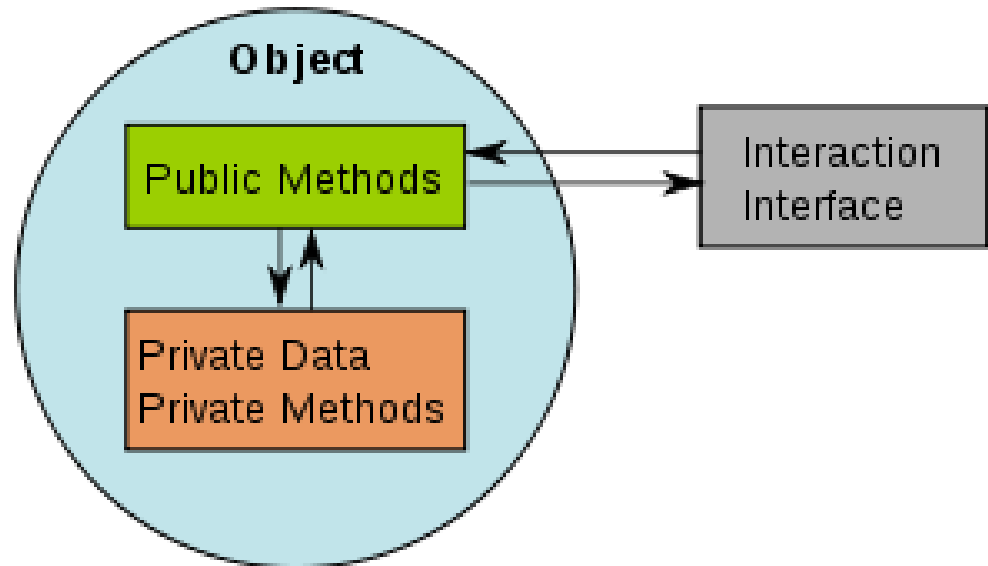
Aula 02 (07/07/2022)



COTI Informática
Escola de Nerds

Encapsulamento

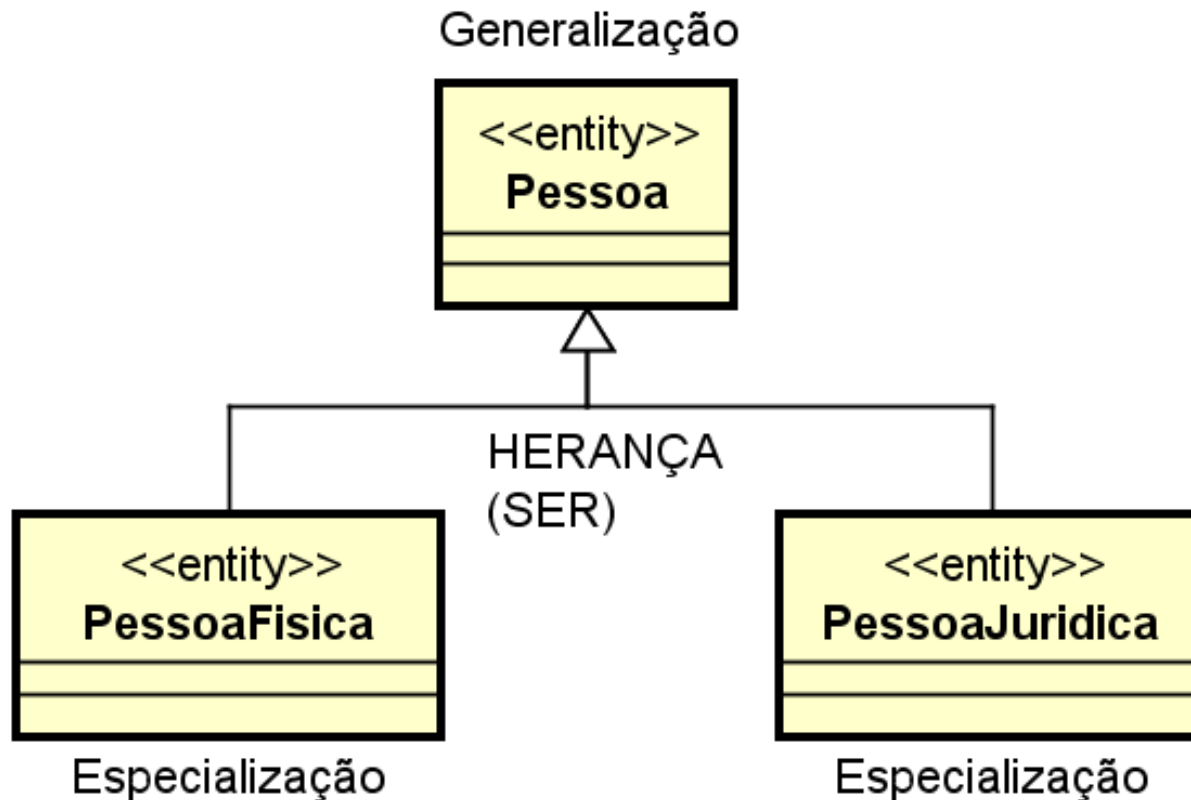
- Ao invés de declararmos os atributos como públicos, iremos mantê-los com visibilidade "private" e criar métodos que permitam acessar os atributos.
- Um exemplo de encapsulamento ocorre quando uma classe declara seus atributos como privados e cria métodos públicos que permitem acessar indiretamente os atributos.
- Estes métodos são chamados de **set** e **get**
- **set** (entrada de dados)
- **get** (saida de dados)



Herança

- É um tipo de relacionamento entre classes que utiliza a abstração do verbo "SER". Define uma relação de hierarquia entre classes, também chamado de generalização / especialização.

Exemplo:

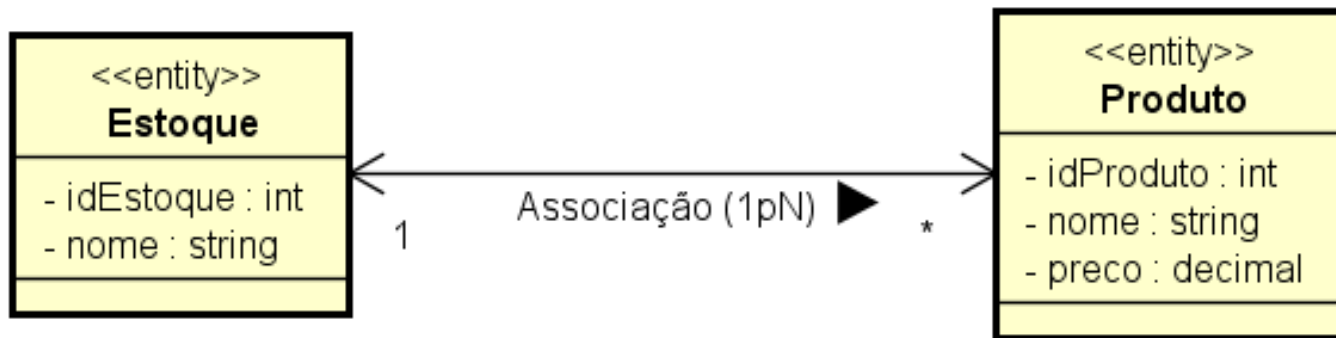


Relacionamento de Associação

Associação (TER)

É um relacionamento baseado na abstração do verbo TER, representa o conceito de utilização (TODO/PARTE) ao invés de herança.

Exemplo:

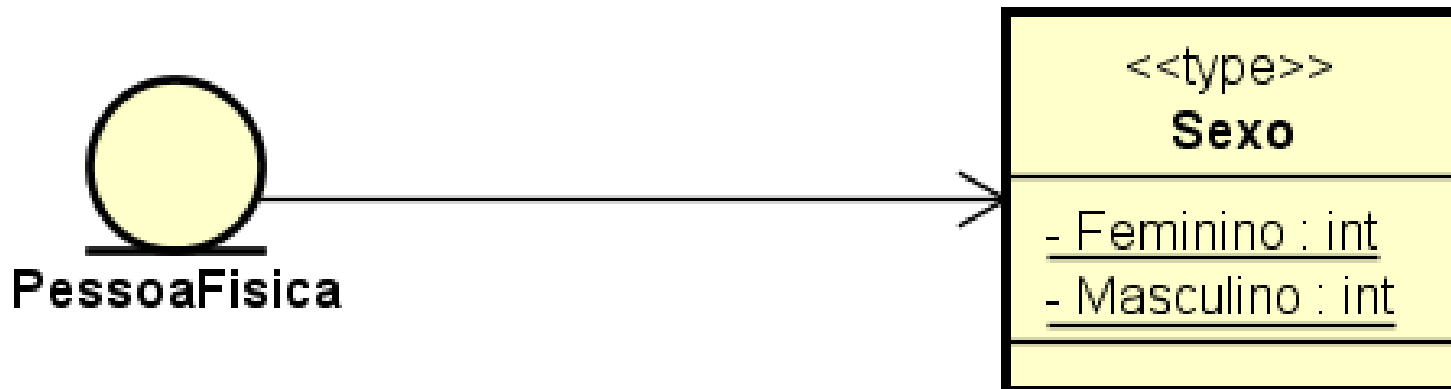


0..1	No mínimo zero e no máximo 1
1	1 e somente 1
0..*	No mínimo zero e no máximo muitos
1..*	No mínimo 1 e no máximo muitos
*	Muitos

ENUMs

- Um tipo de enumeração (também chamado de uma enumeração ou enum) fornece uma maneira eficiente para definir um conjunto de constantes integrais nomeadas que podem ser atribuídas a um valor. Por exemplo, suponha que você precisa definir uma variável cujo valor representará um dia da semana. Há apenas sete valores significativos que essa variável armazenará. Para definir esses valores, você pode usar um tipo de enumeração, que é declarado usando a palavra-chave enum.

Exemplo:



Sobrecarga de Métodos (Overloading)

- Recurso de programação orientada a objetos que permite ao desenvolvedor declarar métodos em uma classe com o mesmo nome, porem com entrada de argumentos diferentes.

Por exemplo:

```
public class Calculo
{
    public double Somar(double a, double b)
    {
        return a + b;
    }

    public double Somar(double a, double b, double c)
    {
        return a + b + c;
    }
}
```

