



Encapsulamento

Programação Orientada a Objetos

Leone Rodrigues Santos

Introdução

- Em OO, é fundamental proteger certas partes do sistema, tais como dados. Para isso, pode-se **esconder** partes do código, não os deixar visíveis para o mundo exterior.
- Isso é fundamental para o controle de modificação em uma classe, principalmente os **atributos**.
- Determinado objeto oferece um serviço, mas quem usa não sabe como ele faz ou mesmo o que ele faz. A única coisa que deve importar para quem usa o serviço é o resultado final.

Encapsulamento

- Tomando como exemplo um uber, não interessa o nome e o CPF do(a) motorista para quem contrata o serviço, apenas importa que o passageiro chegue ao destino desejado.
- Quem está no mundo exterior também não consegue ver quem é o(a) motorista.



Formas de utilizar um método ou atributo

- Dentro da própria classe que o define, por meio de um método que faça a manipulação ou por atribuição.
- Através de um objeto da classe.
- Através de Herança, um dos pilares da Orientação a Objeto.

Modificadores de acesso

- Em linguagens OO, como Java, existem palavras reservadas ou maneiras de declarar atributos e métodos para controlar o acesso a estes.
- Em Java, tem-se as palavras reservadas **public**, **private** e **protected**.
- Em Dart, usa-se um "_" antes do nome do atributo/método. Por exemplo: `_nome`, `_cpf`, `_consultarMatricula()`;
- Por padrão, quando atributos/métodos não estão acompanhados de modificadores de acesso, significa que estão com **visibilidade padrão**. Ou seja, qualquer parte do código pode acessá-los.