



O que é encapsular?



**Certified
Developer**

The Ultimate Tech Degree

DigitalHouse >
Coding School



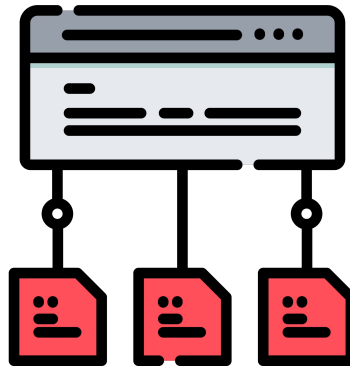
O **encapsulamento** é uma das propriedades mais importantes da programação orientada a objetos.





Encapsulamento

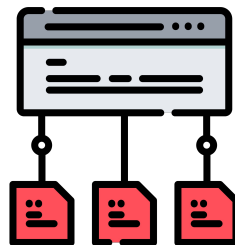
- Quando falamos de encapsulamento, não devemos esquecer a origem da palavra "**colocar nas cápsulas**".
- Quais seriam as cápsulas?
 - É uma embalagem que protege o conteúdo interno. Em POO, procuramos evitar que qualquer outro objeto tenha acesso à estrutura interna de um objeto.





Encapsulamento

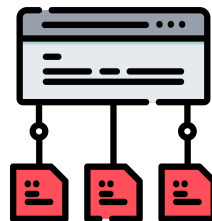
- O conceito de encapsulamento está ligado aos conceitos de **information hiding** (ocultação da informação) e **implementation hiding** (ocultação da implementação).
- Sem encapsulamento qualquer “programa” pode acessar um conjunto de dados.
- O acesso aos dados de uma entidade deve ser restrito e é o desenvolvedor que tem controle sobre quais programas podem alterar quais informações através do uso dos **modificadores de acesso**.





Encapsulamento

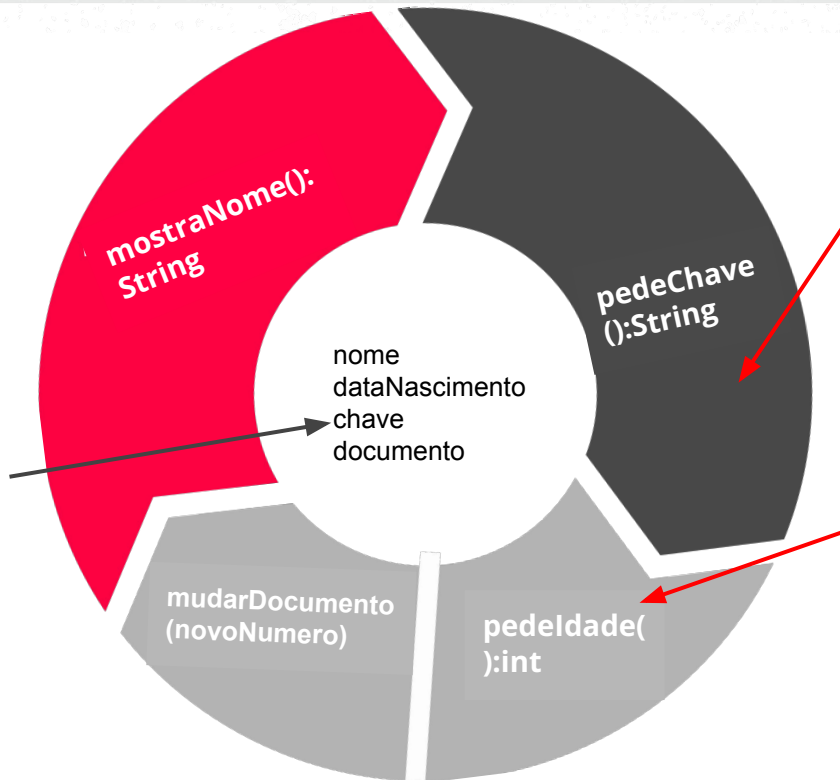
- A partir de agora, quando projetamos nossos objetos, temos que levar em conta o **encapsulamento**.
- Por exemplo, se tivéssemos um objeto **Pessoa** cujo atributo é a chave de acesso do banco, não seria conveniente que todos os objetos pudessem acessar livremente esse atributo.
- O objeto **Pessoa** deve estabelecer um método controlado e seguro para retornar a chave de acesso do banco, por exemplo, se as medidas de segurança forem atendidas.





Pessoa

Ninguém pode ver ou modificar diretamente as informações que formam a estrutura interna do objeto, apenas o próprio objeto.



Para mudar a senha
Eu te peço para fazer isso (dando o PIN).

Para saber a idade, peço que me devolva (Não sei como ele vai calcular).



Importante

- Quando definimos um objeto, deixe seus atributos privados.
- Os métodos que são públicos serão vistos pelos outros objetos.
- Sempre use métodos públicos para visualizar ou modificar as características de seus objetos.
- Para alterar o valor de um atributo, um método **set** é usado, por exemplo, para alterar o nome ele será **setNome (String)**,
- Para obter o valor de um atributo, um método **get** é usado, por exemplo, para alterar o nome ele será **getNome()**.



Os métodos para visualizar ou alterar atributos são chamados de **getters** e **setters**, respectivamente.



DigitalHouse>
Coding School