Encapsulamento focado em Python

**O que é encapsulamento?**

É como colocar os dados de um objeto dentro de uma "caixinha" e controlar quem pode mexer neles. Isso evita bagunça e protege as informações.

**Por que usar?**

Evita que alguém altere dados importantes de forma errada.

Deixa o código mais seguro.

Permite validar ou transformar antes de validar.

**Como funciona em Python?**

**Python usa convenções para indicar o nível de acesso:**

**Nome 🡪público(todos podem acessar)**

**\_nome 🡪protegido(só classe de herdeiros devem usar)**

**\_\_nome🡪privado(só a própria classe acessa)**

**Exemplo simples:**

**class Pessoa:**

**def \_\_init\_\_(self, nome, idade):**

**self.\_nome = nome**

**self.\_\_idade = idade**

**def get\_idade(self):**

**return self.\_\_idade**

**def set\_idade(self, nova\_idade):**

**if nova\_idade >= 0:**

**self.\_\_idade = nova\_idade**

**Descrição do Exemplo**

**No exemplo da classe, uso um modelo básico de uma pessoa com dois atributos: nome e idade.**

**O atributo ­\_nome é protegido, ainda pode ser acessado fora da classe, mas deve ser usado com cuidado.**

**O atributo \_\_idade é privado , o que faz que seja diretamente ser acessado pela própria classe.**