Python

Matheus Andreoli Vieira 2ºY Nº20

**Recurso do encapsulamento em Python**

Uma parte importante da programação orientada ao objeto é o recurso de encapsulamento. Esse recurso envolve agrupar métodos e atributos em pacotes que contém diferentes formas de acesso. Por exemplo, certos métodos e atributos são empacotados como públicos e podem ser acessados por qualquer classe outros pacotes vão ter alguma restrição de acesso – os métodos e atributos só poderão ser acessadas por uma classe filha ou nenhuma outra classe pode acessar.

**Modificadores de acesso e métodos de classe**

## O Python não tem modificador de acesso

O Python não tem o conceito de métodos privados. O que significa que não deve existe restrições arbitrárias ao acessar partes de uma classe. Portanto, não há maneiras de impor um método ou atributo ser estritamente privado. Porém, para se poder ter o recurso de encapsulamento o Python tem um conceito de variáveis ocultas no qual o python não restringe o acesso, mas dificulta o acesso a determinada variável a mutilando. Para o Python tem-se dois termos para classificar as variáveis privados- métodos e atributos fracamente privados e os fortemente privados. A seleção entre esses dois métodos de dificultamento de acesso é determinado inteiramente pelo seu nome.

**Métodos e atributos fracamente privados (\_single\_leading\_underscore)**

As variáveis e métodos que são fracamente privados são os métodos que tem um sublinhado (\_) no início. Esse tipo de variável tem o nome de sublinhado único liderando (em inglês \_single\_leading\_underscore). Essa abordagem APENAS sinaliza que essas variáveis são privadas e outros programadores não devem usar ela em códigos externos.

**Strongly private methods (\_\_double\_leading\_underscore)**

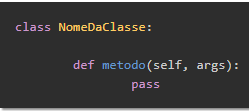
Métodos e atributos fortemente privados tem um duplo sublinhado (\_\_) no início de seus nomes. Esse tipo de variável tem o nome de sublinhado duplo liderando (em inglês \_\_double\_leading\_underscore). Isso causa com que o nome do método seja “mutilado” e não possa ser encontrado/acessado fora da classe -os escondendo. Mas vale ressaltar que isso não para acessar esses métodos que sejam privados, mas para evitar bugs se alguma subclasse tem métodos e atributos com o mesmo nome. O nome “mutilado” ainda pode ser acessado fora da classe.

<https://professormarcolan.com.br/encapsulamento-em-python/#:~:text=O%20Python%20n%C3%A3o%20tem%20modificador%20de%20acesso&text=Por%20exemplo%2C%20em%20C%2B%2B%20o,classe%20pode%20acessar%20os%20elementos.>

**Métodos e atributos em uma classe**

Uma **classe** associa dados (**atributos**) e operações (**métodos**) numa só estrutura. Um **objeto é uma instância** de uma classe. Ou seja, uma representação da classe. Por exemplo, Matheus é uma instância de uma classe chamada Pessoa, mas a Pessoa é a classe que o representa de uma forma genérica. Se você criar um outro objeto chamado João, esse objeto também será uma instancia da classe Pessoa.

Criando uma classe:

****

<http://pythonclub.com.br/introducao-classes-metodos-python-basico.html#:~:text=Segundo%20a%20documenta%C3%A7%C3%A3o%20do%20Python,seja%2C%20uma%20representa%C3%A7%C3%A3o%20da%20classe.>