Trabalho sobre Orientação a Objetos

Aluno - Leonardo Medeiros de Freitas

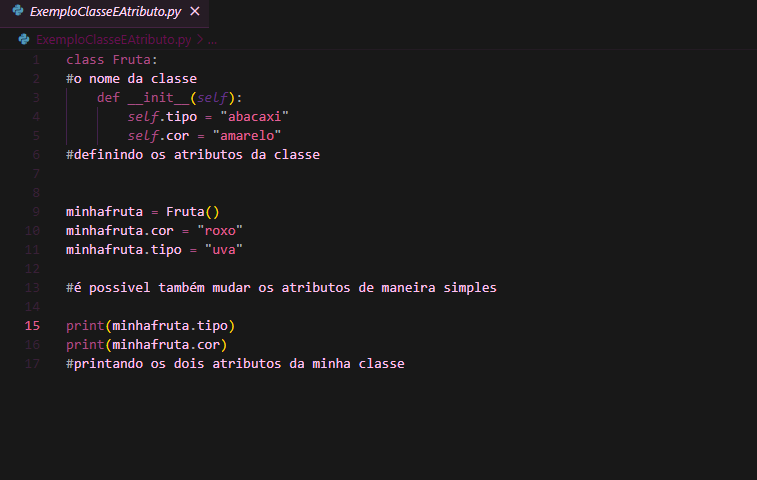
Segundo período de CC.

Classes e Objetos:

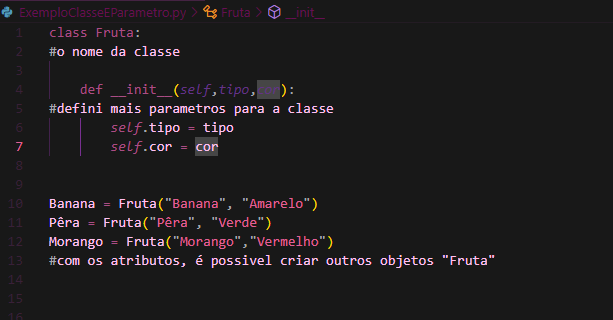
Basicamente, as classes são usadas para organizar dados, funcionalidades, objetos e acessar funções, o que permite serem anexados atributos e parâmetros a elas, para facilitarem a implementação de códigos.

Dentro dos objetos, existem também os atributos e os parâmetros

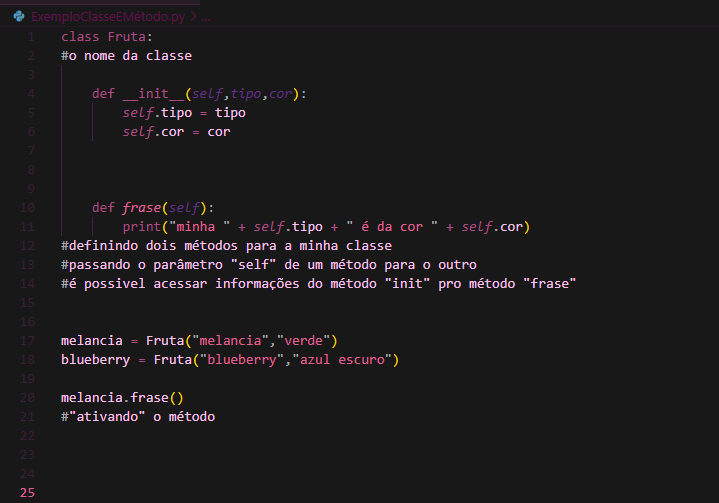
Atributos representam uma “data” ou informações sobre o objeto, como tamanho, cor e velocidade, por exemplo.



Parâmetros representam as tasks que o objeto pode fazer, como aumentar ou diminuir o tamanho e a velocidade e mudar a cor, por exemplo.

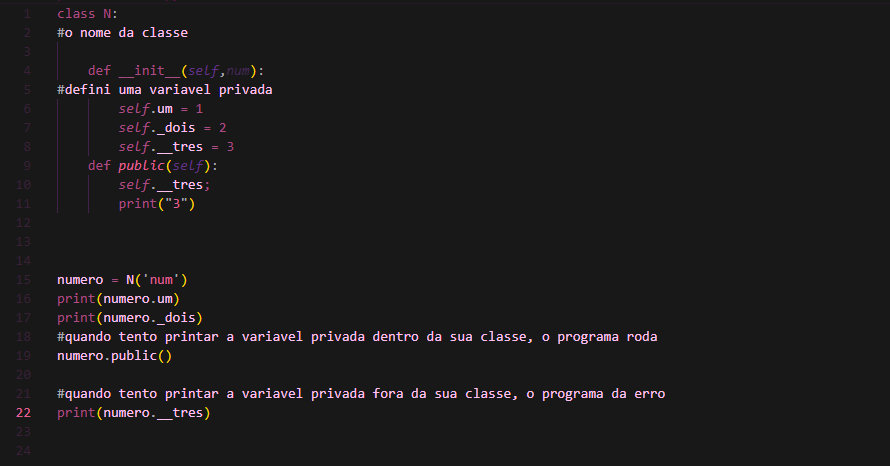


Métodos são como funções relacionadas apenas às classes específicas deles



Tipos de objetos são o que os objetos podem ser ao serem declarados

São eles: Listas, Dicionários, Tuplas. Números, “Strings”, Conjuntos e Arquivos.

Variáveis privadas são aquelas que não podem ser acessadas fora da classe. Elas são escritas com dois (2) underlines (\_\_) na frente delas.

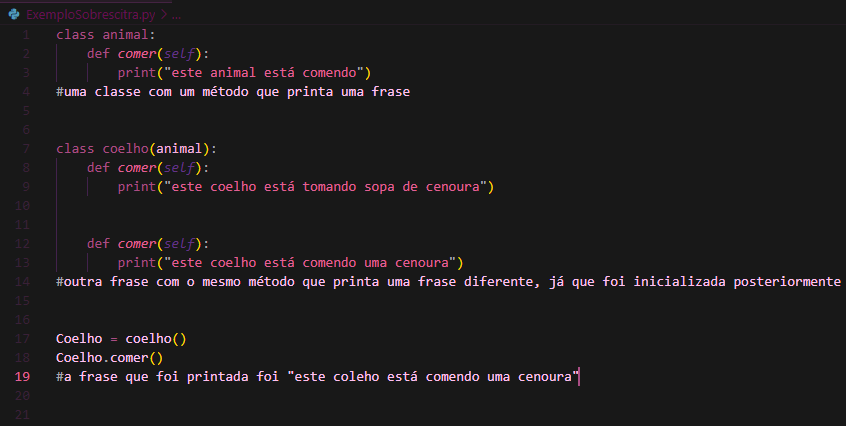
Herança é basicamente quando uma classe herda tudo de outra classe, sem precisar se repetir. Com a Herança, é possível a criação de SuperClasses, que é basicamente a classe principal, e as SubClasses, que são classes menores que possuem a SuperClasse inseridas nelas.

Métodos Herdados são os métodos que são adicionados às subclasses após ela receber uma classe, e que são somados a novos atributos e métodos dessa subclasse que foi criada.

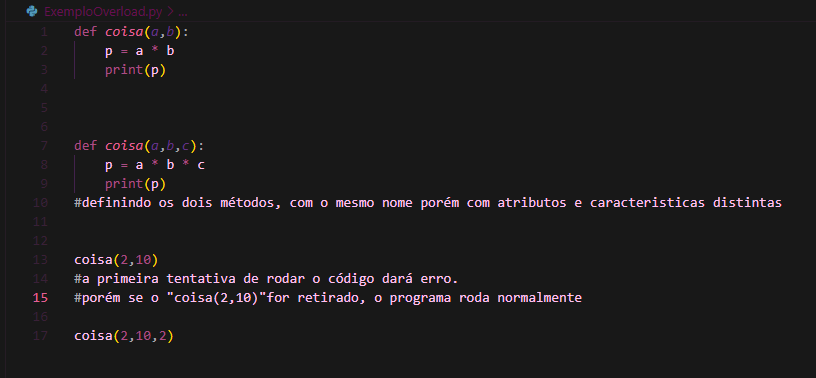
Herança Múltipla consiste, de um modo simples, em uma classe herdar características de várias classes simultaneamente.



Sobrescrita de métodos é declarar uma “def” já declarada anteriormente e mudar as suas características e parâmetros. O programa sempre irá escrever os métodos que estão na parte final do código. Ou seja, se existem três métodos iguais escritos no código, o programa lerá o que foi escrito por último.



Diferentemente de C e Java, o Python não suporta sobrecarga de métodos. É possível sobrecarregar métodos, porém apenas o último método é utilizado no programa.



Interface se trata de um template de classe. Com essas classes, é possível implementar esse template. Isso permite com que as classes que não pertencem a um mesmo tipo ou hierarquia sejam tratadas como se fossem iguais.

Deste modo, a interface diz para o compilador que a classe é um subtipo API a qual foi definida pela própria interface. Assim, ela vira uma uma interface explícita onde há uma interface declarada com um nome específico, onde outras classes são obrigadas a ter esse mesmo nome, se compatíveis.