



O método `__del__`

Assim como precisamos criar um construtor, que irá definir como os objetos instanciados a partir de uma classe serão criados, precisamos definir, também, seu destrutor. Basicamente, um destrutor irá ditar as regras para a destruição dos objetos instanciados a partir de uma classe.

No Python, os destrutores são definidos através do método `__del__` e neles podemos informar a lógica que será executada ao excluir um determinado objeto. Além disso, os destrutores são chamados sempre que um objeto não é mais utilizado em um script, liberando o espaço na memória que este estava utilizando. Abaixo podemos ver como criar um destrutor da classe Pessoa em Python:

```
1 # classe Pessoa
2 class Pessoa():
3     def __init__(self, nome, idade, sexo):
4         self.nome = nome
5         self.idade = idade
6         self.sexo = sexo
7     def __del__(self):
8         print("Destruindo objeto")
```

```
1 #classe main
2 from pessoa import Pessoa
3
4 pessoa1 = Pessoa("João", 25, "Masculino")
5 print(pessoa1.nome) #João #Destruindo objeto
```

Dessa forma, ao finalizar o uso do objeto `pessoa1`, o Python irá invocar o método destrutor para remover o objeto da memória.