

Les classes en Python

Une classe est un ensemble incluant des variables ou attributs et des fonctions ou méthodes. Les attributs sont des variables accessibles depuis toute méthode de la classe où elles sont définies

Histoire

Les origines de la Programmation Orientée Objet remontent au langage de programmation Simula 67, posant les fondations de la POO, telles que les classes, le polymorphisme et l'héritage. Alan Kay créa ensuite réellement la POO avec les objets et l'encapsulation et plus encore.

Définir une classe

```
class Carre:
    """ La classe Carre contient deux méthodes, une pour calculer le périmètre
    d'un carré et l'autre l'aire d'un carré
    l doit être un entier """
    def __init__(self, l):
        self.l = l
    def perimetre(self):
        return 4*self.l
    def aire(self):
        return self.l * self.l
```

Une classe se définit par le mot clé « class » suivi du nom de celle-ci
Le nom doit toujours avoir sa première lettre en majuscule, par convention
Elle ne prend pas de paramètres comme une fonction classique,
Elle est suivie facultativement d'une documentation

Définir une classe

```
class Carre:
    """ La classe Carre contient deux méthodes, une pour calculer le périmètre
    d'un carré et l'autre l'aire d'un carré
    l doit être un entier """
    def __init__(self, l):
        self.l = l
    def perimetre(self):
        return 4*self.l
    def aire(self):
        return self.l * self.l
```

Une fonction contient des méthodes, des fonctions.

Mais sa première fonction est primordiale : l'initialisation (def __init__(**self**, paramètre)) via le constructeur

Elle est obligatoire et impose les paramètres de la classe, self non compris.

Elle attribue les valeurs qu'on a donné a un objet de la classe pour que les fonctions l'utilisent par la suite

Définir une classe

```
class Carre:
    """ La classe Carre contient deux méthodes, une pour calculer le périmètre
    d'un carré et l'autre l'aire d'un carré
    l doit être un entier """
    def __init__(self, l):
        self.l = l
    def perimetre(self):
        return 4*self.l
    def aire(self):
        return self.l * self.l
```

self est aussi par convention le premier paramètre de chaque fonction dans une classe, même si elle ne l'utilise pas.

Les valeurs d'un objet d'une classe doivent être précédée de self.

Tel que self.paramètre

Programmation Orientée Objet

Les classes sont très utilisées dans la P.O.O

L'idée de base de la programmation orientée objet consiste à regrouper dans un même ensemble (l'objet), à la fois un certain nombre de données (ce sont les *attributs d'instance*), et les algorithmes destinés à effectuer divers traitements sur ces données (ce sont les *méthodes*, à savoir des fonctions particulières encapsulées dans l'objet).

Class Véhicule:

Attributs :

Type = « Audi »
Modèle = « 2019 »
Prix = « 99580 »
Couleur = « rouge »

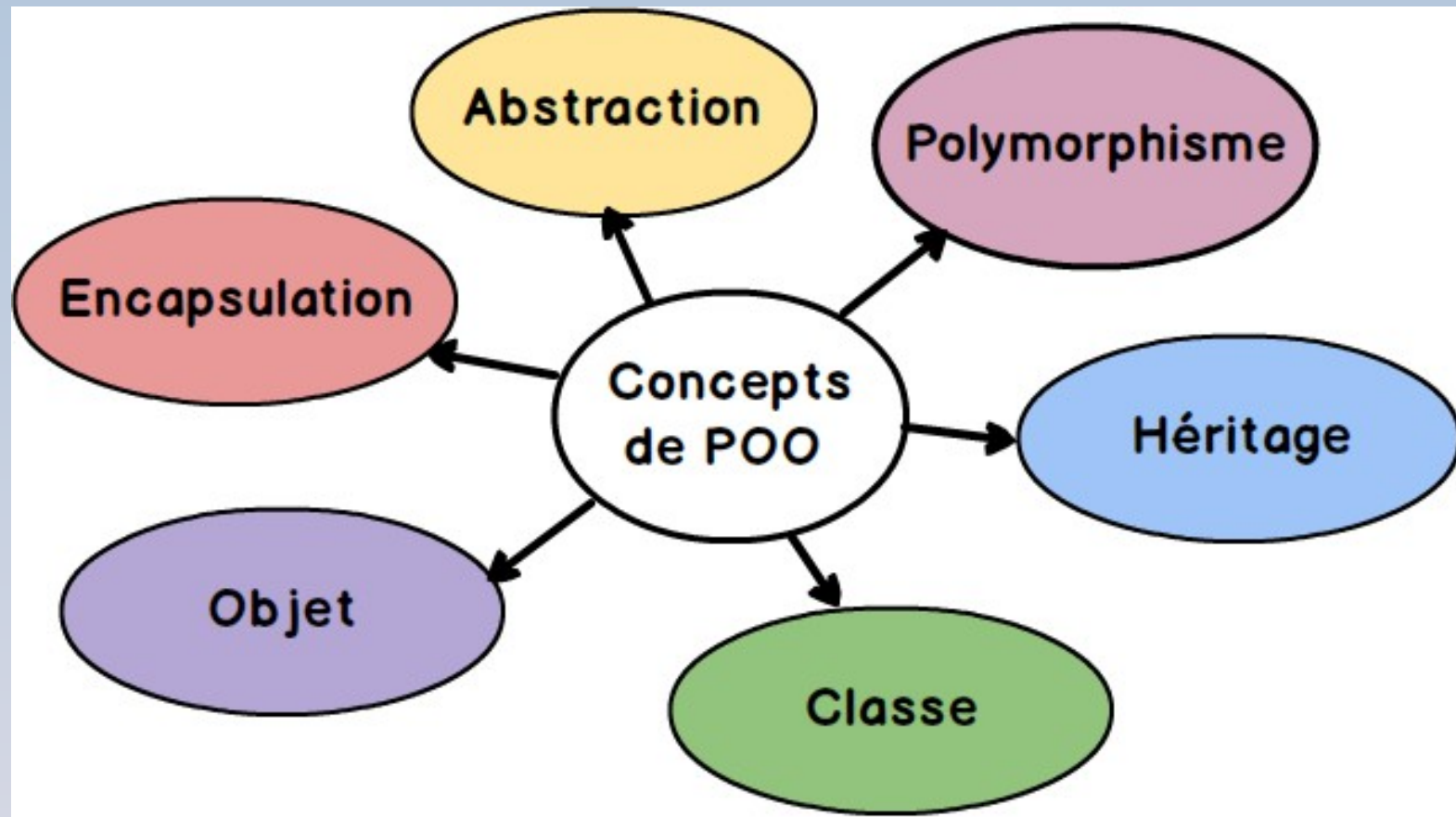
Méthodes :

Def accélérer()
def ralentir()
def tourner()
def freiner()

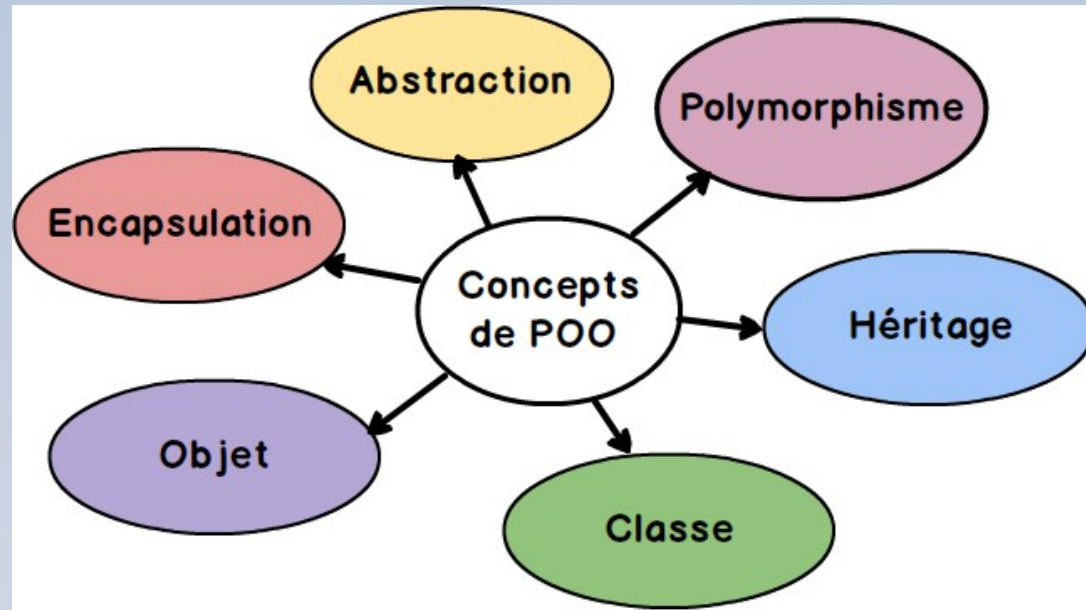


Objet : Voiture





L'héritage consiste à créer des classes qui hériteront des caractéristiques d'une autre classe, elle hérite donc de ses caractéristiques. Plus grossièrement, "des classes dans des classes" (Même si la tabulation des autres classes peuvent ne pas être dans celle de la classe parente)



Supposons :

1- une classe Vehicule avec les attributs : code, prix, marque, couleur, capacité et la méthode AfficherInfos().

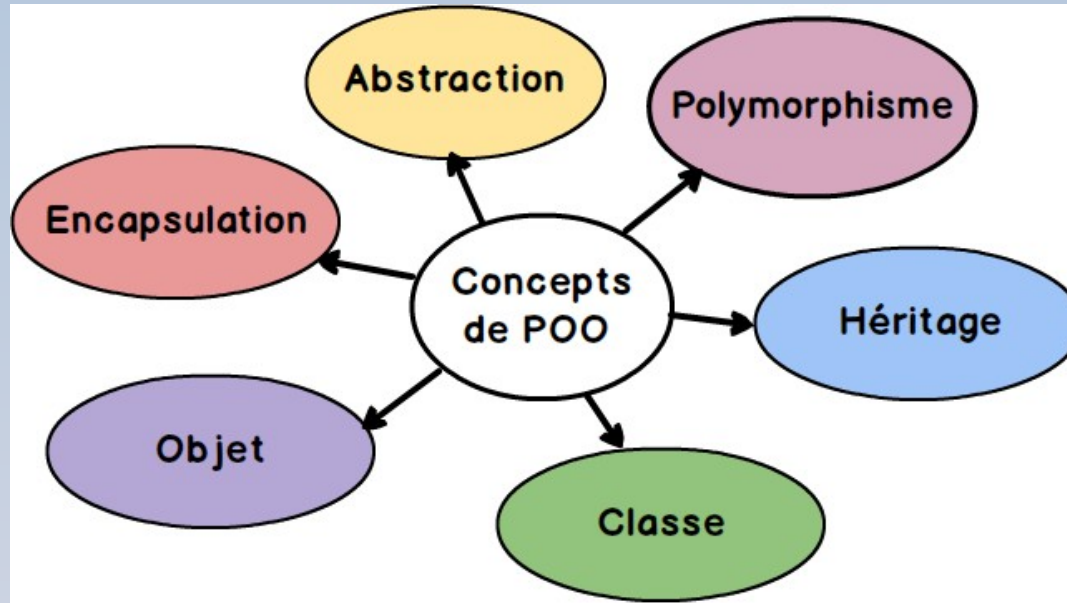
2- On crée deux classes filles de cette classe qui sont :

La classe Moto avec les attributs spécifiques vitesse, modele (plus les autres attributs hérités de la classe Vehicule) et les méthodes : AfficherInfos() , Accelerer(), Arreter(), AfficherModele().

La classe Vélo qui a simplement les éléments (méthode et attributs) hérités.

Supposons qu'on exige de ne jamais avoir des objets de la classe Vehicule. Autrement dit, cette classe est créée pour devenir une mère des classes Vélo et Moto en imposant qu'elle ne sera jamais instanciée. Alors, on dit que cette classe doit être une classe abstraite.

Une classe est dite abstraite, si elle ne peut être instanciée. Une classe est abstraite si elle a au moins une méthode abstraite. Elle met en scène donc une classe mère et fille, où la classe mère ne doit pas être instanciée par l'utilisateur.



L'encapsulation désigne le regroupement de données avec un ensemble de méthodes qui en permettent la lecture et la manipulation.

Les propriétés des objets ne peuvent être accédées que par ses méthodes. Ainsi, la classe encapsule à la fois les attributs et les méthodes qui permettent de manipuler les objets indépendamment de leurs états.

Ainsi, l'encapsulation est un mécanisme qui empêche donc de modifier ou d'accéder aux objets par un autre moyen que les méthodes proposées, et de ce fait, permet de garantir l'intégrité des objets.

Déclarer une classe

```
<tmp 3>
1 class Carre:
2     """ La classe Carre contient deux méthodes, une pour
3     calculer le périmètre
4     d'un carré et l'autre pour l'aire du même carré
5     l est la longueur d'un côté, ici un nombre entier"""
6
7     def __init__(self, l):
8         self.l = l
9
10    def perimetre(self):
11        return 4*self.l
12
13    def aire(self):
14        return self.l * self.l|
```

```
Shells
Python Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) on Windows (64 bits).
This is the Pyzo interpreter with integrated event loop for ASYNCIO.
Type 'help' for help, type '?' for a list of *magic* commands.

>>> (executing file "<tmp 3>")

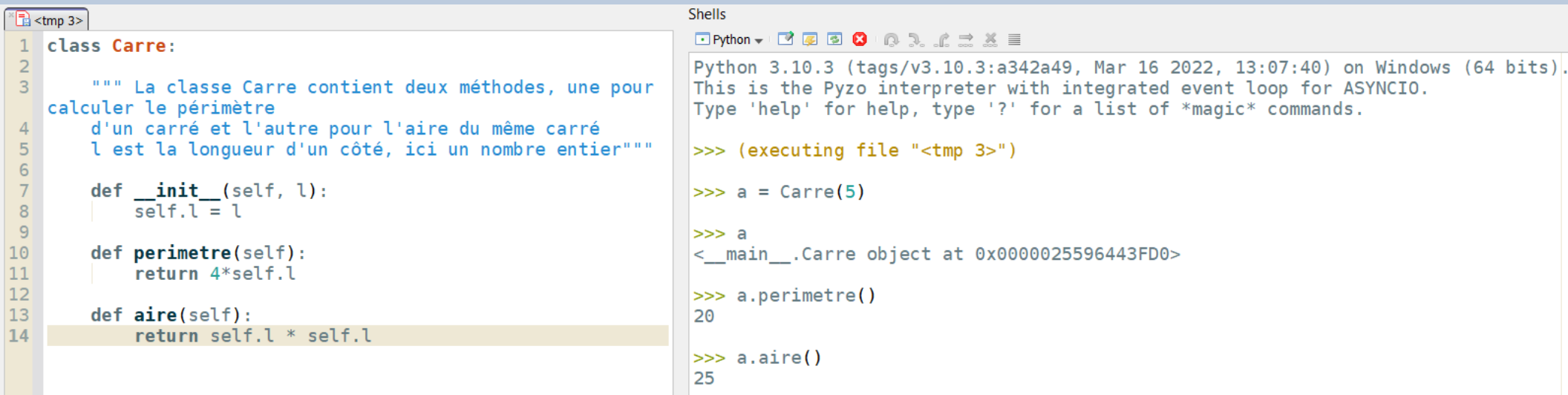
>>> a = Carre(5)

>>> a
<__main__.Carre object at 0x0000025596443FD0>

>>>
```

Ici, a est désormais un objet de la classe Carre. Nous avons décidé de lui donner la valeur 5. Nous pouvons bien voir que a est un objet de la classe Carre et qu'elle a sa place dans la mémoire.

Utilisation



```
1 class Carre:
2     """ La classe Carre contient deux méthodes, une pour
3     calculer le périmètre
4     d'un carré et l'autre pour l'aire du même carré
5     l est la longueur d'un côté, ici un nombre entier"""
6
7     def __init__(self, l):
8         self.l = l
9
10    def perimetre(self):
11        return 4*self.l
12
13    def aire(self):
14        return self.l * self.l
```

```
Shells
Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) on Windows (64 bits).
This is the Pyzo interpreter with integrated event loop for ASYNCIO.
Type 'help' for help, type '?' for a list of *magic* commands.

>>> (executing file "<tmp 3>")

>>> a = Carre(5)

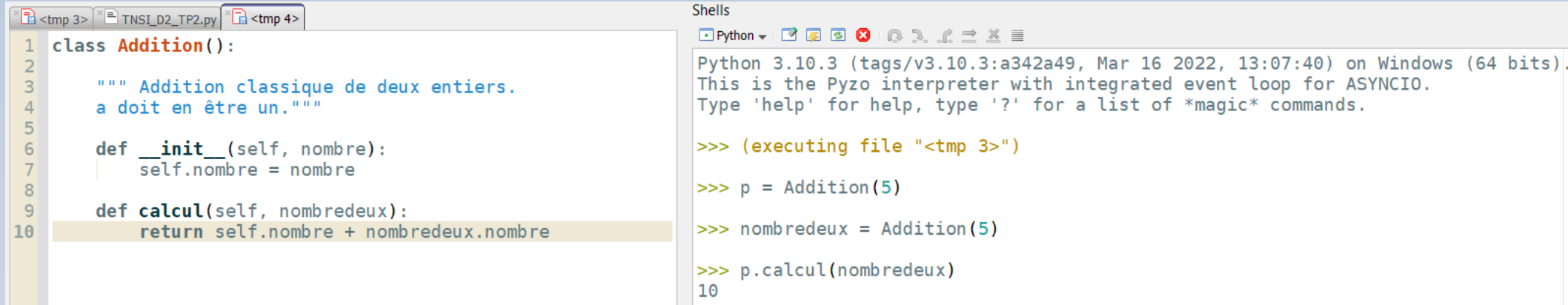
>>> a
<__main__.Carre object at 0x0000025596443FD0>

>>> a.perimetre()
20

>>> a.aire()
25
```

Nous pouvons ensuite exécuter les méthodes de la classe, précédée du nom de l'objet de la classe.

Utilisation



The image shows a Python IDE with two main panes. The left pane displays a Python class named `Addition` with two methods: `__init__` and `calcul`. The right pane, titled 'Shells', shows the execution of the code in a Python 3.10.3 environment. The execution steps are as follows:

```
Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) on Windows (64 bits).  
This is the Pyzo interpreter with integrated event loop for ASYNCIO.  
Type 'help' for help, type '?' for a list of *magic* commands.  
  
>>> (executing file "<tmp 3>")  
  
>>> p = Addition(5)  
  
>>> nombredeux = Addition(5)  
  
>>> p.calcul(nombredeux)  
10
```

```
1 class Addition():  
2  
3     """ Addition classique de deux entiers.  
4     a doit en être un."""  
5  
6     def __init__(self, nombre):  
7         self.nombre = nombre  
8  
9     def calcul(self, nombredeux):  
10        return self.nombre + nombredeux.nombre
```

Les méthodes peuvent bien sûr appeler d'autres paramètres.

Merci de votre écoute

Sources :

<https://docs.python.org/3/tutorial/classes.html>

<https://waytolearnx.com/2020/01/quest-ce-que-la-poo.html>

<https://www.9raytifclick.com/cours/abstraction-en-poo-et-implementation-en-python/>

<https://python.developpez.com/tutoriels/composition-programmes/abstraction-donnees/?page=programmation-orientee-objet>

<https://fr.acervolima.com/encapsulation-en-python/>

<https://www.data-transitionnumerique.com/encapsulation-poo>

<https://youtu.be/dQw4w9WgXcQ>