

# Python

## Attributs et méthodes de classe

### Méthodes statiques

Nicolas Delestre

# Attributs de classe

## Déclaration, utilisation

- On déclare un attribut de classe en dehors de toute méthode, en ne le préfixant pas par `self` :
  - il est souvent initialisé juste après la déclaration de la classe
  - il est utilisé dans les méthodes de classe ou les méthodes d'objet

## point.py

```
5 class Point2D:
6     _les_points = []

16     def __init__(self, x: float, y: float, identifiant: str=None):
17         self._x = x
18         self._y = y
19         self._id = identifiant
20         self._les_points.append(self)
```

## Déclaration

- On déclare une méthode de classe :
  - en préfixant la déclaration du décorateur `@classmethod`
  - en utilisant uniquement des attributs de classe
  - en remplaçant le `self` par `cls` (bonne pratique)

## point.py

```
5 class Point2D:  
6     _les_points = []  
7  
8     @classmethod  
9     def les_points(cls):  
10         return cls._les_points
```

## Utilisation

- On utilise une méthode de classe en préfixant l'appel de la méthode par le nom de classe ou par un objet de la classe (notation pointée)

```
1 >>> from point import Point2D
2 >>> Point2D.les_points()
3 []
4 >>> pt1=Point2D(1,1)
5 >>> Point2D(2,2)
6 Point2D(2, 2)
7 >>> Point2D.les_points()
8 [Point2D(1, 1), Point2D(2, 2)]
9 >>> pt1.les_points()
10 [Point2D(1, 1), Point2D(2, 2)]
```

## Déclaration

- On déclare une méthode statique en préfixant la déclaration du décorateur `@staticmethod`
- Une méthode statique n'a pas accès à l'état de la classe, il s'agit d'une fonction utilitaire rattachée à la classe (attention aux faux amis avec d'autres langages tels que le Java)

## point.py

```
12 @staticmethod
13 def translation(pt, vecteur, id=None):
14     return Point2D(pt.x + vecteur.x, pt.y + vecteur.y, id)
```

## Utilisation

- On utilise une méthode statique en préfixant l'appel de la méthode par le nom de classe ou un objet de la classe (notation pointée)

```
1 >>> from point import Point2D
2 >>> from vecteur import Vecteur2D
3 >>> Point2D.translation(Point2D(1,1), Vecteur2D(1,0,'v'))
4 Point2D(2, 1)
```

# Conclusion

---

- Attributs et méthodes de classes
- Méthodes statiques
- Utilisable depuis la classe ou depuis une instance de la classe