4.1 Introduction

NSI Tle - JB Duthoit

4.1.1 Qu'est ce que la POO ?

Considérons un objet de la vie de tous les jours : un smartphone par exemple, que l'on a ouvert pour voir ses entrailles.
Il est évident qu'en regardant cet objet, on est frappé par sa complexité (pour un non spécialiste).
Par contre si l'on le referme, l'utilisateur pourra utiliser les fonctionnalités du smartphone (grâce aux boutons, aux applications diverses...) sans pour autant en comprendre le fonctionnement (qui est très complexe).

En programmation objet, on va faire un peu le même chose ; créer des objets pour ensuite les utiliser sans se soucier de leur complexité interne.

La programmation orientée objet (POO) , est qu'il existe des milliers d'objets prêts à être utilisés, et vous en avez déjà utilisé de nombreuses fois sans le savoir. Par exemple, lorsque'en Python on utilise des chaînes de caractères ou des listes, on travaille avec des objets !

☞ Étude de la <class list> !

Les idées sous-tendant le paradigme objet datent des années 60. Mais il faudra attendre le début des années 70 et la mise au point du langage Smalltalk pour que le paradigme objet gagne en popularité chez les informaticiens. Aujourd'hui de nombreux langages permettent d'utiliser le paradigme objet : C++, Java, Python...

4.1.2 Conception d'un objet

Voici 5 étapes (utiles pour un débutant) pour établir une conception orientée objet :

- Identifier les objets et leurs attributs : on cherche à identifier les objets du monde réel que l'on souhaiterait réaliser.
- Identifier les opérations : quels sont les actions que l'objet pourrait subir de la part de son environnement et qu'il peut provoquer sur son environnement.
- Établir la visibilité : Quelles relations avec les autres objets ?
- Établir l'interface. Cette interface définit précisément quelles fonctionnalités sont accessibles, et sous quelles formes.
- Implémenter les objets : on écrit le code.

4.1.3 Les classes

Une ***classe*** est une description d'un ensemble d'objets ayant une structure de données commune (***attributs***) et pouvant réaliser des actions (***méthodes***) .

On considère une classe comme un nouveau type de données. On appelle ***instance*** de la classe l'objet qui la représente.

On peut représenter une classe comme ceci :

Nom de la classe
Attributs : - Attribut1 - Attribut 2 ....
Méthodes - Méthode1() - Méthode2() ....