

Programmation impérative vs POO

Oui, nous sommes bien dans le chapitre concernant la programmation impérative et pourtant, nous allons parler "objet" !

Pourquoi ? Tout simplement parce qu'en Python, tout (ou presque) est objet.

Programmation impérative

La programmation impérative est un paradigme de programmation qui utilise des instructions pour modifier l'état d'un programme. En programmation impérative, le programmeur décrit explicitement les étapes que l'ordinateur doit suivre pour obtenir le résultat souhaité. Cela implique d'écrire une séquence de commandes que l'ordinateur exécute dans l'ordre.

Pour simplifier, on fait souvent une comparaison à une "recette" de cuisine. Il s'agit d'écrire toutes les étapes pour fabriquer une pizza (par exemple)

La programmation impérative s'oppose à la **programmation déclarative**, dans laquelle le programmeur spécifie quel est le résultat souhaité et l'ordinateur détermine comment y parvenir.

Pour continuer avec la métaphore de la pizza, il s'agira maintenant d'appeler une pizzeria et de déclarer que vous souhaitez obtenir une pizza 4 fromages !

Programmation Orientée Objet

C'est un passage obligé lors de l'apprentissage de nombreux langages informatiques et est également un sujet relativement ardu lorsqu'on nous le présente pour la première fois.

Il est possible que vous ne compreniez pas tout d'un coup : pas d'inquiétude, suivez le cours à votre rythme et sachez que nous reviendrons sur ces concepts dans le chapitre b.POOP

Qu'est-ce que la programmation orientée objet ?

La programmation orientée objet (ou POO en abrégé) correspond à une autre manière d'imaginer, de construire et d'organiser son code.

La programmation orientée objet repose sur le concept d'objets qui sont des entités qui vont pouvoir posséder un ensemble de variables et de fonctions qui leur sont propres.

Les objectifs principaux de la programmation orientée objet sont de nous permettre de créer des scripts plus clairs, mieux structurés, plus modulables et plus faciles à maintenir et à déboguer.

Le Python, un langage orienté objet

Python est un langage résolument orienté objet, ce qui signifie que le langage tout entier est construit autour de la notion d'objets.

En fait, quasiment tout en Python est avant tout un objet et nous allons manipuler des objets dès le début de ce cours

Pour apprécier Python et utiliser toutes ses fonctionnalités, il est donc indispensable de comprendre cette composante orienté objet.

Qu'est-ce qu'un objet en programmation ?

Dans la vie réelle, un objet possède des caractéristiques et nous permet de réaliser des actions. Un crayon par exemple possède une taille, une couleur, une forme, etc. qui sont ses caractéristiques et permet d'écrire ou de dessiner.

Le concept d'objets en informatique s'inspire fortement de cette définition de la vie réelle : on va appeler "objet" un bloc cohérent de code qui possède ses propres variables (qui sont l'équivalent des caractéristiques des objets de tous les jours) et fonctions (qui sont nos actions). Comme les objets de la vie courante, les objets informatiques peuvent être très simples ou très complexes.

Standards de code

<https://peps.python.org/pep-0008/>