

LA PROGRAMMATION ORIENTÉE OBJET

La programmation orientée objet est **une méthode de programmation informatique de plus en plus plébiscitée, que ce soit dans le développement logiciel ou la data science.**

Organisée autour des objets, ou données, la programmation orientée objet offre de nombreux avantages.

La programmation orientée objet (POO) est un paradigme informatique consistant à définir et à faire interagir des objets grâce à différentes technologies, notamment les langages de programmation (Python, Java, C++, Ruby, Visual Basic.NET, Simula...). **On appelle objet, un ensemble de variables complexes et de fonctions, comme par exemple un bouton ou une fenêtre sur l'ordinateur, des personnes (avec les noms, adresse...), une musique, une voiture... Presque tout peut être considéré comme un objet.** L'objectif de la programmation orientée objet est de se concentrer sur l'objet lui-même et les données, plutôt que sur la logique nécessaire et les actions à mener pour faire cette manipulation.

POO : Comment ça se passe ?

La programmation orientée objet se déroule en plusieurs étapes. **La première consiste à modéliser les données en identifiant les objets que le programmeur souhaite manipuler ainsi que leurs interactions.** Une fois que les objets ont été modélisés, **ils sont ensuite conceptualisés dans une classe d'objets, qui aura des attributs et des méthodes. Les attributs correspondent aux caractéristiques de l'objet et les méthodes aux actions que peut faire cet objet.**

En masquant les fonctions et variables de l'objet, le programmeur crée un code source parfois complexe mais facilement utilisable, tout en renforçant la sécurité du système et en évitant la corruption accidentelle de données. Grâce à ses propriétés d'héritage et de polymorphisme, une classe est réutilisable par le programme pour lequel elle a été créée, mais aussi par d'autres programmes orientés objet.

Les concepts clés de la POO

- **La classe** : une classe est un ensemble de code contenant des variables et des fonctions permettant de créer des objets. Une classe peut instancier ou générer plusieurs objets.
- **Les objets** : un objet est un bloc de code mêlant des variables et des fonctions, appelées respectivement attributs et méthodes. Les attributs définissent les caractéristiques d'un objet ou d'une classe, les méthodes définissent quant à elles les fonctions propres aux instances d'une classe.
- **L'encapsulation** : l'encapsulation permet d'enfermer dans une capsule les données brutes afin d'éviter des erreurs de manipulation ou de corruptions des données. L'encapsulation permet ainsi de cacher des méthodes et des attributs à l'extérieur de la classe.

- **L'abstraction** : le concept d'abstraction consiste à cacher les détails inutiles à l'utilisateur final d'une classe. Il va ainsi pouvoir utiliser une classe dans son code de programmation tout en ne sachant pas comment celle-ci a été développée.
- **L'héritage** : le concept d'héritage signifie qu'une classe B va hériter des mêmes attributs et méthodes qu'une classe A. Lorsqu'une instance de la classe B est créée, on peut alors appeler les méthodes présentes dans la classe A par la classe B. Cela va permettre de faire gagner du temps au programmeur.
- **Le polymorphisme** : lorsqu'une classe hérite des méthodes d'une classe parent, il est possible de surcharger une méthode, qui consiste à redéfinir la méthode de la classe parent pour que les deux classes ne fassent pas les mêmes tâches.

TP sur la POO