**Design Pattern**

1. **Pattern Strategie:**

**Le pattern Stratégies** définit une famille d’algorithmes, encapsule chacun d’eux et les rend interchangeables.

* Il permet à l’algorithme de varier indépendamment des clients qui l’utilisent.

Ici **héritage(EST UN)**  n est pas vraiment la solution pour une modification du code ou une maintenance, la solution se trouve dans **interface(A UN)**  qui va permettre de créer des types nouveaux.

* Son fonctionnement est le suivant : **repérez tous les comportements d’une classe qui varie et créer des interface qui contiendront ses comportements et par la suite en fonction de nos besoins définir des classes qui implémenterons ses interfaces et définiront ses comportements**