





**SOLID**

ORDER-NCLOSEDPRINTABLE

MAGNET 2-DUALS - THOUGHTS

**« Software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification », *Bertrand Meyer***

**« You should be able to extend the behavior of a system without having to modify that system. », *Uncle Bob***

**Indissociable du S, cela se traduit dans les logiciels par:**

- **La définition de contrats publics / implémentations privées**
- **L'extension grâce au polymorphisme : sous-typage, paramétrique (a.k.a générique) ou ad-hoc**

INDÉPENDANTES

# SOLID

## OPEN-CLOSED PRINCIPLE

« Software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification », *Bertrand Meyer*

« You should be able to extend the behavior of a system without having to modify that system. », *Uncle Bob*

Indissociable du S, cela se traduit dans les logiciels par:

- La définition de contrats publics / implémentations privées
- L'extension grâce au polymorphisme : sous-typage, paramétrique (a.k.a générique) ou ad-hoc

## INDÉPENDANT DES IDIOMES



# SOLID

## (BARBARA) LISKOV SUBSTITUTION PRINCIPLE

« *Subtype Requirement*: Let  $\phi(x)$  be a property provable about objects  $x$  of type  $T$ . Then  $\phi(y)$  should be true for objects  $y$  of type  $S$  where  $S$  is a subtype of  $T$ . », *Barbara Liskov*

Cette propriété vise à garantir l'interopérabilité sémantique des types dans une hiérarchie de type:

- La définition de contrats publics / implémentations privées
- L'extension grâce au polymorphisme : sous-typage, paramétrique (a.k.a générique) ou ad-hoc