

## INTRODUCTION :

Les design patterns ou patrons de conception en français sont des solutions génériques à des problèmes spécifiques. On dit qu'ils sont spécifiques car c'est aux développeurs de coder cette solution. Il en existe plusieurs, mais nous nous pencherons vers le pattern observer ou patron d'observation.

## DEFINITION :

Le pattern Observer est un design pattern qui définit une relation un à plusieurs entre des objets. Le pattern observer est un modèle de conception utilisé dans la programmation de logiciel orienté objet pour créer des relations entre objets de manières dynamiques.

## PRINCIPE :

Le principe de base du pattern Observer est de créer une relation de dépendance un-à-plusieurs entre un **objet** (appelé **sujet** ou observable) et plusieurs autres objets (appelé **observateurs**). Les observateurs eux sont dépendants du sujet, c'est-à-dire quand l'état du sujet change, ces derniers sont informés et peuvent être mis à jour. En d'autres termes le sujet maintient une liste des observateurs et les informe automatiquement de tout changement d'état. Les observateurs eux peuvent être mis à jour et effectuer des actions en réponse à ces changements. Il est à noter que ce pattern observer est une implémentation possible du **OCP (Open/Closed Principle)**, en utilisant le pattern observer, nous pouvons ajouter de nouveaux observateurs pour réagir aux changements du sujet sans avoir à modifier le sujet lui-même ; et il implémente aussi le **DIP (Dependency Iversion Principle)**, en utilisant ce pattern, les observateurs dépendent du sujet, plutôt que l'inverse. Cela signifie que le sujet ne connaît pas directement ces observateurs et n'a pas besoin de les modifier ou de les connaître quand il change. Au lieu de cela, il envoie simplement des informations aux observateurs qui sont liés à lui.