

TP 7

Design Pattern "Observer"

Dans ce TP, nous allons mettre en place le design pattern « Observer ».

Petit Rappel :

On utilise ce design pattern quand on doit gérer des événements.

Dans une classe qui doit déclencher des événements, on ajoute:

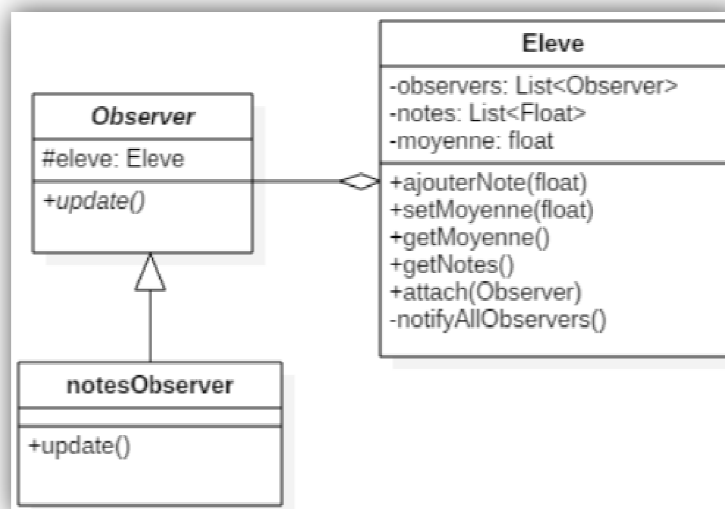
- En attribut : une liste d'Observateurs
- Une méthode permettant d'ajouter un Observateur dans la liste
- Une méthode permettant d'envoyer un signal à tous ses observateurs.

Observateur est une classe abstraite avec une méthode signal, dont héritent des observateurs "concrets" qui implémentent cette méthode.

Quand l'état de la classe change, elle doit envoyer un signal à tous ses observateurs qui doivent effectuer l'action nécessaire en fonction du nouvel état de la classe.

Ci-après le diagramme de classes que nous allons réaliser.

Dans ce diagramme, la classe qui déclenche les événements est **Eleve**.



Etape 1 :

On crée une classe abstraite **Observer** avec une méthode abstraite **update()**.

Etape 2 :

On crée la classe **Eleve** en initialisant ses deux paramètres "*observers*" et "*notes*" dans son constructeur sans paramètres.

On implémente toutes ses méthodes en respectant les fonctionnalités suivantes :

- Lorsqu'on ajoute une nouvelle note, on doit notifier tous les observateurs en invoquant la méthode prévue pour cela.
- notifier tous les observateurs (envoyer le signal) \Rightarrow invoquer la méthode *update* pour tous les observateurs.
- Attacher un observateur \Rightarrow l'ajouter à la liste des observateurs.

Etape 3 :

On crée la classe **NotesObserver** qui, dans son constructeur possédant un seul paramètre **Eleve** :

- initialisera le paramètre eleve avec le paramètre d'entrée.
- Attachera (ajoutera) à cet eleve la classe courante **NotesObserver** comme observateur.

Puis on implémente la méthode *update()* :

```
float moyenne = 0;

for(float note : eleve.getNotes()) {
    moyenne += note;
}

moyenne /= eleve.getNotes().size();

eleve.setMoyenne(moyenne);
```

Etape 4 :

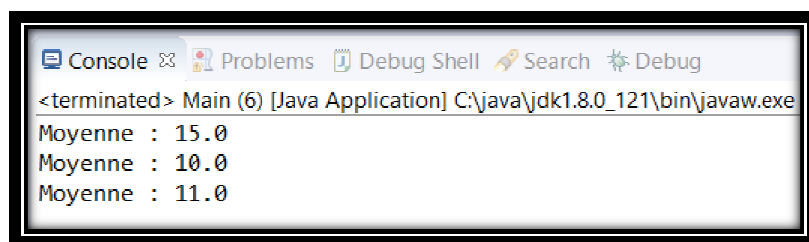
On crée la classe **Main**.

```
Eleve eleve = new Eleve();

new NotesObserver(eleve);

eleve.ajouterNote(15.0f);
System.out.println("Moyenne : " + eleve.getMoyenne());
eleve.ajouterNote(5.0f);
System.out.println("Moyenne : " + eleve.getMoyenne());
eleve.ajouterNote(13.0f);
System.out.println("Moyenne : " + eleve.getMoyenne());
```

Exécutez et observez le résultat !



The screenshot shows a Java IDE console window with the following output:

```
<terminated> Main (6) [Java Application] C:\java\jdk1.8.0_121\bin\javaw.exe
Moyenne : 15.0
Moyenne : 10.0
Moyenne : 11.0
```