

Analyse et Conception Orientée Objet – ING1

TD 11 : Design Pattern

L'objectif de ce TD est de savoir utiliser un patron de conception comme solution à un problème logiciel donné.

Exercice 1. La pluie et le beau temps !!

Dans une application de collecte des indicateurs météorologiques, On propose les classes suivantes :

- `WeatherServer` : capte la température et l'ensoleillement ("tempête", "pluie", "nuageux", "nuages épars", "éclaircies", "soleil"). On simulera ce serveur avec des méthodes
 - o `void setTempérature (int temperature)` pour saisir la température à un instant donné
 - o `int getTemperature ()` pour la récupérer.
 - o `void setEnsoleillement (String ensoleillement)` pour saisir l'ensoleillement à un instant donné
 - o `int getEnsoleillement ()` pour le récupérer.
- `SimpleWeatherClient` :
 - o s'enregistre auprès du `WeatherServer`
 - o imprime sur la console la température et l'ensoleillement chaque fois qu'il y a un changement.
- `SuperWeatherClient` :
 - o s'enregistre aussi auprès du `WeatherServer`
 - o imprime sur la console l'heure, la date, la température et l'ensoleillement chaque fois qu'il y a un changement, accompagné d'un smiley :-) s'il fait beau, :-(dans les autres cas.

Chaque client (chaque `SimpleWeatherClient` et chaque `SuperWeatherClient`) porte un nom. Le message qu'il imprime est toujours précédé de son nom.

Exemple de sortie sur la console :

```
simpleclient1 : 25°, soleil
```

```
monSuperClient34 : 25/02/2012, 10h30: 25°, soleil, :-)
```

Question 1 : Quel design pattern a été utilisé ? Illustrez-le par un diagramme de classe. Quel rôle joue la classe `SuperWeatherClient` dans ce pattern ?

Question 2 : Illustrez par un diagramme de séquence l'affectation d'une nouvelle valeur de température captée ?

Question 3 : Réaliser cette application en java ☺