

TD3

Singleton

Soit une classe `Logger` permettant de *logger* des messages sur la console :

```
public class Logger {
    private int nbLoggedMessages;
    public Logger() {
        this.nbLoggedMessages = 0;
    }
    public void log(String msg) {
        this.nbLoggedMessages++;
        System.out.println("LOG" + this.nbLoggedMessages + ": " + msg);
    }
}
```

Nous souhaitons l'utiliser pour réaliser un système de *logging* centralisé pour une application. Les services de ce système sont fonctionnellement uniques au sein de l'application et doivent pouvoir être appelés par toutes les couches de l'application (en effet, il serait peu pratique de passer une référence au service à toutes les classes devant l'utiliser).

1. Modifier cette classe en utilisant le pattern *Singleton*, afin de garantir qu'une unique instance de la classe sera créée, tout en offrant un point d'accès global à cette instance.
2. Deux types d'implémentation de *Singleton* existent, selon que l'unique instance est créée lors de son premier accès (*lazy instantiation*) ou bien à l'initialisation de la classe (*eager instantiation*). Écrivez l'autre implémentation.
3. Comment faire évoluer le pattern *Singleton* en *Multiton* afin de gérer plusieurs instances ? Proposez une implémentation.

Façade

1. Expliquez en quoi la classe `javax.swing.JOptionPane` est une façade ?
2. Récupérez l'archive `security.jar`, puis importez-la dans Eclipse. Cette archive contient quatre classes (`Alarm`, `Camera`, `Light` et `Sensor`) représentant différents composants d'un système de sécurité (leur implémentation est très sommaire).
 - (a) Écrivez une classe qui utilise directement les classes de l'archive pour créer un système de sécurité composé de cinq capteurs, une alarme, deux caméras et trois lumières. En particulier, la classe contient une méthode pour activer le système de sécurité (*i.e.*, activation des capteurs et des alarmes, mise en route des caméras et extinction des lumières) et une autre pour le désactiver (*i.e.*, désactivation des capteurs, arrêt de la sonnerie des alarmes et extinction des caméras).
 - (b) Écrivez une façade pour simplifier l'utilisation des composants d'un système de sécurité. En particulier, la façade doit permettre de facilement activer et désactiver un système de sécurité.
 - (c) Écrivez une nouvelle classe qui manipule un système de sécurité en utilisant la façade.