

Projet Programmation, Génie Logiciel, Preuves

2015/2016

Note :

- **La bonne documentation de votre code et le respect des conventions du code JAVA seront pris en compte.**
- **Vous créez un projet maven et vous créez un package pour chaque exercice.**

Exercice 1

Nous voulons filtrer des messages entrants selon plusieurs critères, par exemple par type. Le but est de supprimer les messages remplissant les critères souhaités par l'utilisateur de la liste des messages entrants. Les critères de filtrage peuvent évoluer. Un message a un type (qui peut être TEXT, BYTE ou XML), une taille et un contenu.

1. Proposez une solution respectant OCP pour filtrer les messages par
 - par type
 - par taille
 - par mots clés
2. Proposez quelques méthodes de test (JUnit) pour vérifier le bon fonctionnement de votre proposition.

Exercice 2

Nous souhaitons développer une application de dessin pour les enfants. L'application devra permettre les manipulations suivantes :

- Création des formes de base (des cercles, des carrés, rectangles et triangles), les colorer et calculer leur superficie. Lors de la création, on peut spécifier certains paramètres (par exemple rayon et position du centre pour le cercle), si l'utilisateur n'a pas spécifié un des paramètres, l'application lui assigne une valeur par défaut.
 - Modification des formes existantes (changer de couleur, modification de position ou, pour certaines formes, tourner la forme avec un degré donné)
 - Création d'un dessin à partir d'un ensemble de formes, l'utilisateur peut modifier, ajouter ou supprimer ces formes.
 - Le dessin peut inclure une nouvelle forme en spécifiant sa forme (par exemple « cercle » pour inclure un cercle).
 - La couleur des formes peut être donnée avec des valeurs RGB (Red, Green, Blue) ou HSB (Hue, Saturation, Brightness)
 - Les informations sur les formes existantes sont archivées et affichées, l'application doit donc informer les parties responsables de ces tâches à chaque changement effectué sur les formes.
1. Proposez une implémentation permettant ces fonctionnalités. Utilisez un Singleton pour représenter le programme principal.
 2. Retournez un document dans lequel vous expliquez quels sont les design patterns que vous avez utilisés et pourquoi vous les avez utilisés.
 3. Proposez quelques méthodes de test (JUnit) pour vérifier le bon fonctionnement de votre proposition.