

TD/TP4

PATRONS DE CONCEPTION – DECORATEUR

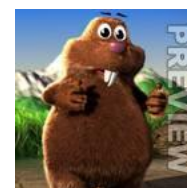
1. EXEMPLE UTILISANT DECORATEUR

Les fichiers des classes de l'exemple du cours¹ figurent partiellement dans le fichier **cafe.zip** sur arche. Faire un projet sous Eclipse et y inclure les fichiers dans un package *cafe*.

- 1.1. Faire le diagramme des classes de ce projet. Observer la factorisation du code en comparaison avec l'exemple du cours.
- 1.2. Ajouter la classe **Deca**, le cout associé est de 1 euro. Ajouter aussi comme décorateur la classe **BoissonCreme**, dont le cout ajouté est de 55c.
- 1.3. Ajouter la méthode **public String toString()** (choisir correctement la classe où placer cette méthode) qui retourne le cout et la description du produit choisi selon l'affichage *description : cout euros*.
- 1.4. Créer une classe exécutable **TestCafe** qui crée un Deca avec de la crème et double portion de chantilly puis l'affiche.
- 1.5. Faire le diagramme de séquence de l'affichage du cout d'un expresso avec chantilly.

2. GENERATEUR DE LOGOS

Une entreprise de création d'applications mobiles vous embauche pour un stage d'été et on vous demande de reprendre un travail commencé par un employé parti en vacances. L'objectif du programme est de fabriquer à la volée des logos de *René la Taupe* qui sont, après demande par sms, envoyés aux utilisateurs. L'archive **logo.zip** vous est fournie.



- 2.1. Ouvrir l'archive que vous trouverez sur arche et visualiser dans un nouveau projet les classes **Rene** et **Main** avec Eclipse (la classe *MyImage* ne vous intéresse pas, son code ne devra pas évoluer). Les images du programme sont dans le fichier **im.zip**, placer le répertoire décompressé à la racine du projet. Lancer ce programme. Puis, dans le fichier *Main.java*, activez l'instruction pour placer un chapeau sur René.
- 2.2. Les lunettes de soleil (cf répertoire *im*) doivent être positionnées aux coordonnées (255,76), écrivez le code permettant d'afficher des lunettes sur René.

¹ **Design Patterns**, Collection Tête la Première, E. Freeman & al., O'Reilly

- 2.3. On souhaite maintenant produire différents logos de *René*, avec un chapeau, ou avec des lunettes, avec un bâton de sucrerie (position (441,202)), avec un smiley (position(260,210)), Ces logos seront produits à la volée, on veut un mécanisme efficace pour afficher différents logos de René facilement.

Vous devrez générer facilement un René avec une combinaison d'accessoires. Il faudrait que les autres équipes du projet puissent générer ces nouveaux logos en une seule ligne de code.

Proposer un diagramme de classes pour résoudre ce problème, le faire valider par votre enseignant.

- 2.4. Codez la solution que vous proposez.
- 2.5. Implémentez aussi une méthode **prix** qui permet de calculer le prix de tout un ensemble de logos. Chaque accessoire a un prix, que l'on ajoute au prix de René.
- 2.6. L'équipe marketing du projet vous demande d'ajouter le même type de logo mais avec une image de *Crazy Frog*. Pouvez-vous facilement modifier votre code pour produire les deux séries de logos ?
- 2.7. Suite au succès rencontré auprès des utilisateurs, l'équipe marketing du projet souhaite ajouter une nouvelle fonctionnalité qui permettrait de mettre un texte, donné par l'utilisateur, sur le logo créé. Modifiez votre diagramme de classe pour ajouter cette fonctionnalité, **le faire valider par votre enseignant**. Codez votre solution (la méthode **public void** textOver(String txt, **int** x, **int** y) de la classe MyImage permet d'écrire un texte dans l'image courante à la position (x,y), elle modifie l'image courante).
- 2.8. Question optionnelle : Imaginez une interface graphique permettant de construire un logo à partir de René la Taupe grâce à des boutons, l'appui sur un bouton correspondra à l'ajout d'un accessoire sur le logo en cours.