

# *Projet de Programmation en Java*

## *Dessine moi un mouton*

### *"Comment devenir un Petit Prince Java"*

L'objectif est de créer un programme fonctionnel en Java utilisant les concepts de la P.O.O.

L'application devra réaliser une grande fresque murale réalisée à partir de plusieurs dessins géométriques.

- Les formes géométriques élémentaires disponibles sont les lignes, les polygones, les cercles, et les ellipses.
- Une image est composée de formes géométriques et peut contenir des images de plus petite taille (aire).  
Aucun doublon de forme géométrique ne peut exister dans une image.
- Un dessin est composé d'images.  
Aucun doublon d'image ne peut exister dans un dessin.
- Les calculs du périmètre et de l'aire des formes géométriques, des images et des dessins sont requis.  
L'aire d'un polygone quelconque peut être calculée par triangulation.  
Le périmètre (l'aire) d'une image est la somme des périmètres (des aires) de ses composants.  
Il en est de même pour un dessin.
- Les formes géométriques peuvent subir les transformations suivantes : homothétie, translation, rotation, symétrie centrale, symétrie axiale.  
Appliquer une transformation sur une image revient à appliquer la même transformation sur ses composants.  
Appliquer une transformation sur un dessin revient à appliquer la même transformation sur ses composants.
- Le tri des formes d'une image selon plusieurs critères: leur périmètre, leur aire et leur distance à l'origine.
- Le clonage d'un dessin est obtenu par clonage de tous ses composants.

Penser à mettre en place une gestion d'exceptions lorsque cela s'avère nécessaire.

Indication: les critères de comparaison pourront être mis en oeuvre via les interfaces Comparable<T> et Comparator<T>.

La mise en oeuvre des points suivants est demandée :

- une **application console** créant la fresque et réalisant une batterie de tests les plus exhaustifs possibles.  
Plus précisément, l'ensemble des services proposé sera testé et leurs résultats affichés (méthodes de calcul de périmètre et d'aire, de transformations, de tri et de clonage).  
La fresque devra être affichée sous une forme textuelle la plus explicite possible.
- une **application graphique 2D** sous *Swing* permettant la demande des mêmes services que l'application console, la réalisation interactive de la fresque, la visualisation d'images représentant les membres du binôme.

## **Lectures conseillées**

<http://imss-www.upmf-grenoble.fr/prevert/Prog/Java/swing/Graphics2D.html>

<http://imss-www.upmf-grenoble.fr/prevert/Prog/Java/swing/image.html>

<http://docs.oracle.com/javase/tutorial/2d/index.html>

## **Rendu du projet**

Le projet est à réaliser par binôme.

Le rendu est sous la forme **NOM1\_NOM2.zip**, archive contenant les fichiers *dessin.jar* et *rapport.pdf*.

Le fichier *dessin.jar* doit contenir les sources Java et la Javadoc.

*Une documentation des sources doit être générée automatiquement (utiliser les annotations adéquates en java afin de fournir une Javadoc).*

Les projets sont à déposer sur l'espace de rendu de *Campus Efrei*, ceux rendus uniquement par mail seront pénalisés.

Merci de votre sens des responsabilités.

Afin de bien mettre en évidence les relations entre les classes de votre projet, vous pouvez utiliser un logiciel qui vous générera un diagramme de classes à partir de vos sources.

L'idéal est d'utiliser un logiciel libre permettant la rétro-conception.

StarUML : <https://www.projet-plume.org/mots-cles-proposes-par-lauteur/retro-ingenierie>

BOUML : <https://www.projet-plume.org/fiche/bouml>

## **Un rapport doit fournir votre analyse et votre conception.**

Les éléments suivants doivent notamment y figurer :

- une introduction exposant clairement les objectifs, limites, choix du projet
- un mode d'emploi de l'application
- des schémas décrivant l'architecture fonctionnelle
- des schémas décrivant les structures de données utilisées
- une explication en français et/ou pseudo-code et/ou langage d'implémentation choisi des principales fonctions (code à fournir et à commenter)
- une conclusion résumant le travail effectué et ouvrant des perspectives
- une bibliographie utilisée
- une table des matières

Les classes conçues doivent être détaillées aussi bien leur interface que leur implémentation.

Une bonne idée est de réaliser le rapport en même temps que le développement.

## **Une présentation de votre projet aura lieu lors de la dernière séance de TP.**