

Tp Dessin avec java et swing

Objectif : programmer une petite application de dessin vectorielle en java.

Technologie : java (v8 ou ...) et api "swing" .

Structure imposée :

une hiérarchie de classes d'objets graphiques à dessiner :

Figure (abstract) avec opérations abstraites (ex : dessiner(Graphics g) , ...)
et

Cercle , **Rectangle** , **Ligne** héritant de Figure et avec coordonnées spécifiques.
polymorphisme attendu autour de dessiner(Graphics g) .

Une classe **FenetrePrincipale** (héritant de **JFrame**) et comportant la méthode **main** .

Cette fenêtre principale comportera (au minimum) 2 sous panneaux organisés en BorderLayout (un au NORTH , un au CENTER) .

Dans le premier sous panneau (JPanel) on placera des composants permettant de

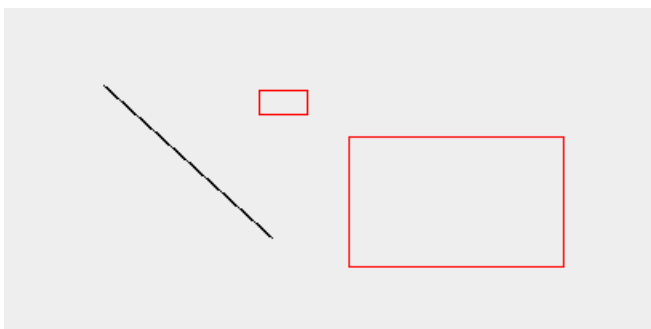
- choisir le type de figures à dessiner (ligne ou rectangle ou cercle)
exemple : 3 boutons radios exclusifs
- choisir la couleur des figures à dessiner (ex : liste déroulante)
- effacer toutes les figures (ex : bouton poussoir)



- ActionListener et actionPerformed() pour tous les boutons
- ItemListener et itemStateChanged() pour la liste déroulante.

NB : certaines actions événementielles agiront (indirectement) sur certains éléments de la classe MyCanvas ci après .

Dans le second panneau (classe **MyCanvas** héritant de JPanel à coder) , on dessinera des figures en y gérant des événements "souris".



--> MouseListener (avec ou sans MouseAdapter)
et mousePressed() et mouseReleased()

NB : La classe MyCanvas (où l'on dessine des figures) devra idéalement comporter les attributs suivants (avec get/set):

- typeFigure
- couleurCourante
- figureCourante
- listeFigures (List<Figure>)

.../...

Améliorations facultatives :

On pourra éventuellement imaginer et programmer tout un tas d'améliorations vis à vis des fonctionnalités déjà citées :

- dessin temporaire (en mode XOR) lorsque le bouton de la souris n'est pas encore relâché
- mise en place de certains "design patterns" (ex : visitor)
- ...

Partie facultative du Tp :

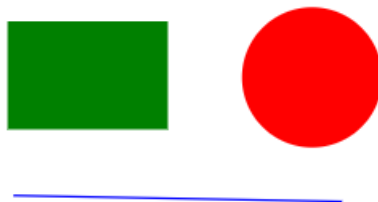
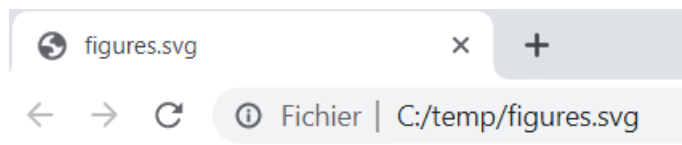
générer un fichier "fig.svg" à partir des coordonnées des figures dessinées et stockées dans listeFigures .

Ce fichier "fig.svg" ou "figures.svg" devra pouvoir être visualisé via un navigateur internet récent .

Déclenchement : menu ou bien bouton poussoir "export svg" .

Conseils :

- opération abstraite (polymorphe) ".toSvgString()" sur Figure et sous classes .
- utiliser MessageFormat.format(...)
- FileOutputStream , PrintStream , ...



Exemple de syntaxe svg (à générer) :

```
<svg xmlns='http://www.w3.org/2000/svg' width='600' height='400'>
  <rect x='146' y='40' width='80' height='54' style='fill:green' />
  <circle cx='298' cy='68' r='35' style='fill:red' />
  <line x1='149' y1='127' x2='313' y2='130' style='stroke:blue' />
</svg>
```