

TP 3 Java

Java.awt.geom.* contient plusieurs classes relatives aux primitives géométriques (comme: **Point2D**, **Rectangle2D**, **Ellipse2D**, **Line2D**, **Area** et même **Path2D** ou **GeneralPath** –ces derniers permettent de tracer une forme graphique généralisée-).

La classe **Area** permet de combiner des formes géométriques simples pour avoir des formes plus complexes (**Area.add**, **intersect**, **subtract** et **exclusiveOr**).

Le code suivant permet de combiner une ellipse avec un rectangle pour obtenir une forme plus complexe (dans la fonction **paint** ou une fonction générée **dessin**):

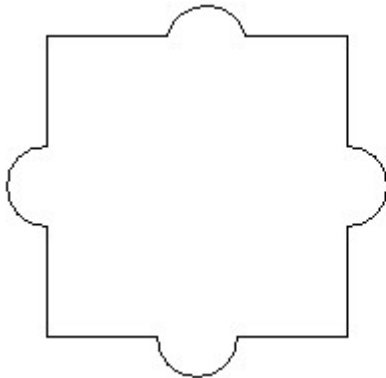
```
Graphics2D g2 = (Graphics2D) g;  
Ellipse2D oval = new Ellipse2D.Float(100, 200, 80, 60);  
Rectangle2D rect = new Rectangle2D.Float(120, 230, 90, 80);  
Area forme = new Area(oval);  
forme.add(new Area(rect)); // ou forme.intersect(new Area(rect));  
// ou forme.subtract(new Area(rect));  
  
g2.draw(forme);  
  
AffineTransform at = new AffineTransform();  
at.translate(100, 100);  
at.rotate (0.5);  
forme.transform(at);  
g2.draw(forme); // ou g2.fill(forme);
```

Pour spécifier l'épaisseur de trait du dessin (ex: 3): **g2.setStroke(new BasicStroke(3.0f));**

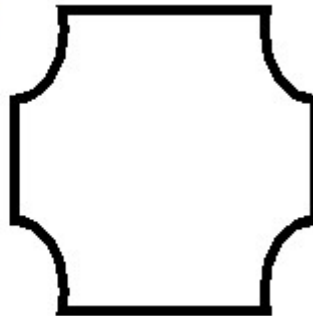
Exercice 1:

1) Ecrire le code java (avec Area ...) qui permet d'afficher les formes suivantes (carrés de longueur et largeur 200, avec des cercles ou ellipses sur les côtés):

1)



2)



2) Utiliser une boucle pour effectuer une série de rotations animées des formes 1 et 2 autour de leur centre par pas d'angle de 15 degrés environ.

Exercice 2 :

Charger une image et l'afficher, ensuite:

1) faire une rotation de l'image autour de son centre d'un angle de 45 degrés. Se servir de l'exercice 2 pour la rotation.

(Utiliser les instructions suivantes: avec **import java.awt.Image, java.awt.Toolkit**
image= Toolkit.getDefaultToolkit().getImage (fichier_image); et **Graphics2D.drawImage**
(image, x, y, largeur, hauteur, this);).

2) Faire un changement d'échelle sur l'image (valeur de 2 sur x, et 1.5 sur y)