

# TP1

## Documentation de l'API Java

La documentation de l'API officielle de Java se trouve sur le site :

<http://java.sun.com/javase/6/docs/api/>

Prenez le temps de vous familiariser avec cette documentation et n'hésitez pas à utiliser les fonctions de recherche de votre navigateur (ctrl+f) si vous cherchez une classe en particulier. Elle est organisée comme suit :

- en haut à gauche : liste de tous les packages
- à gauche en dessous : liste de toutes les classes
- à droite : documentation de la classe sélectionnée

## Hello world

Essayez de coder et d'exécuter chacun des petits programmes *Hello world* ci-dessous.

1. Le programme Hello world en mode console qui a été vu en TD.
2. Pour créer une applet qui fait la même chose, on utilise le fichier source suivant :  
Il faut ensuite le compiler pour obtenir du byte-code et créer un fichier `html` qui appelle l'applet :

```
import java.awt.*;
import javax.swing.*;
public class MonApplet extends JApplet {
    public void paint(Graphics g) {
        g.drawString("Hello world!",20,20);
    }
}
```

Le fichier `html` doit ensuite être ouvert avec votre navigateur ou avec l'application `appletviewer`.

```
<html>
<head>
  <title>Mon applet</title>
</head>
<body>
  Mon applet :
  <applet code="MonApplet.class" width="90" height="90"></applet>
</body>
</html>
```

3. Enfin la version suivante permet d'afficher le texte dans une fenêtre.

```
import javax.swing.JOptionPane;
public class HelloWorldDiag {
    public static void main(String[] args) {
        JOptionPane.showMessageDialog(null, "Hello world!");
        System.exit(0);
    }
}
```

## Applet de dessin

Les méthodes de dessins sont à rechercher dans la description de la classe `Graphics` du package `awt`. Reprenez la structure du code source et du fichier html de l'exercice précédent et modifiez les pour construire une applet qui dessine les formes simples suivantes :

- un segment est défini par ses extrémités  $A$  et  $B$  :  
`drawLine(int,int,int,int)` qui prend en paramètres les coordonnées  $x_A, y_A, x_B, y_B$ .
- un rectangle est défini par son coin supérieur gauche  $A$ , sa largeur  $l$  et sa hauteur  $h$  :  
`drawRect(int,int,int,int)` et `fillRect(int,int,int,int)` qui prennent en paramètres  $x_A, y_A, l, h$ . `fillRect` remplit le rectangle avec la couleur courante.
- un polygône est défini par les coordonnées de ses sommets et son nombre de sommets :  
`drawPolygon(int[],int[],int)` et `fillPolygon(int[],int[],int)` qui prennent en paramètres la liste des coordonnées  $x$ , la liste des coordonnées  $y$ , et le nombre de sommets. `fillPolygon` remplit le polygône avec la couleur courante.
- une ellipse est dessinée en fonction du rectangle dans lequel elle est inscrite :  
`drawOval(int,int,int,int)` et `fillOval(int,int,int,int)` prennent les même paramètres que `drawRect`.
- pour sélectionner une nouvelle couleur :  
`setColor(Color c)` prend en argument un objet `Color` dont les champs sont par exemple `black`, `green`, `blue`, etc...

Faites un programme qui dessine automatiquement la figure ci-dessous, puis faites de même pour dessiner cette fois-ci un damier.

