

# Applets (classes Java2D) : applet

## Floride2D

### Introduction

En Java 2, Java2D remplace les classe Graphics de Java 1. Java2D est plus performant et gère les couleurs.

### Convertir un objet en 2D

Pour créer des graphiques avec Java 2D, on déclare la méthode paint() de l'applet comme avant, mais dans les instructions qui accompagnent cette méthode, on doit convertir l'objet Graphics en objet Graphics2D.

- `public void paint ( Graphics screen ) {`
- `Graphics2D screen2D = (Graphics2D) screen;`
- `// suite des instructions`
- `}`

### Couleurs

Pour colorer quelque chose, on doit d'abord créer un objet color et ensuite attribuer cet objet à la chose que l'on veut colorer.

- `Color c1 = new Color(0.807F, 1F, 0F) ; // nombres à virgule flottante`
- `Color c2 = new Color(255, 204, 102); // nombres normaux`
- `screen2D.setColor(Color.yellow); // couleur basique prédéfinie`

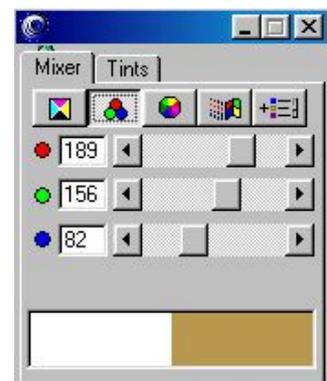
La 3<sup>ème</sup> instruction indique que toutes les opérations de dessin se feront dans la couleur définie (ici, le jaune). Pour modifier les couleurs de fond d'écran (background) et de dessin (foreground), on utilise ces méthodes :

- `setBackground(Color.white) ;`
- `setForeground(Color.black);`

Voici un tableau avec les principales couleurs:

Nom de la couleur en Java	Valeur RVB	En français
black	(0,0,0)	Noir
blue	(0,0,255)	Bleu
cyan	(0,255,255)	Cyan (bleu primaire)
darkGray	(64,64,64)	Gris foncé
gray	(128,128,128)	Gris
green	(0,255,0)	Vert
lightGray	(192,192,192)	Gris clair
magenta	(255,0,255)	Magenta (rouge primaire)
orange	(255,200,0)	Orangé
pink	(255,175,175)	Rose
red	(255,0,0)	Rouge (scarlet)
white	(255,255,255)	Blanc
yellow	(255,255,0)	Jaune

Les couleurs RVB représentent les valeurs entre 0 et 255 de rouge, vert et bleu. En fonction de leur concentration, les couleurs se nuancent. La capture d'écran de Macromedia Freehand illustre le choix d'une couleur Camel (189,156,82).



## Dégradés de couleurs

Le titre est très parlant...

- `GradientPaint(x1,y1,Color1,x2,y2,Color2,boolean)` ;

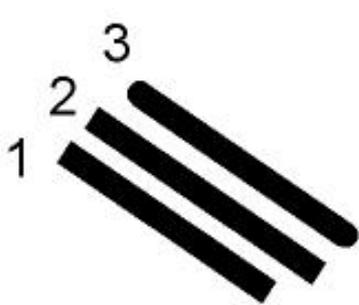
x1,x2 et Color1 représentent les coordonnées et la couleur du point de départ du dégradé.

x2,y2 et Color2 représentent les coordonnées et la couleur du point d'arrivée du dégradé.

Boolean est true si le dégradé est cyclique, sinon, il n'est pas nécessaire de le préciser.

## Définir un type de trait

Une des nouveautés de Java2D est la possibilité de faire de beaux traits. Ce qui est beau, ce sont les extrémités ou les raccords entre 2 traits.



### Traits :

- 1) CAP\_BUT
- 2) CAP\_SQUARE
- 3) CAP\_ROUND

### Jointures

- 1) JOIN\_MITER
- 2) JOIN\_ROUND
- 3) JOIN\_BEVEL

Vous remarquerez qu'il existe une différence de longueur entre les traits 1 et 2. C'est l'unique différence entre CAP\_BUT (le classique) et CAP\_SQUARE.

Pour créer un trait, il faut utiliser la méthode `setStroke()` de l'objet `BasicStroke()`. Elle admet 3 arguments :

- Valeur *float* pour la largeur de la ligne (en pixel).
- Valeur *int* pour le style d'ornement, extrémité de ligne.
- Valeur *int* pour le style d'ornement, jonction entre 2 lignes.

- `BasicStroke pen = BasicStroke (2.0f, BasicStroke.CAP_BUTT, BasicStroke.JOIN_ROUND);`
- `screen2D.setStroke(pen);`

## Créer des objets à dessiner

Les arguments sont les mêmes que pour des objets classiques (allez donc voir le chapitre Graphics pour plus d'informations).

Tous ces objets font partie de `java.awt.geom`

### Lignes :

`Lines2D.Float ligne = new Line2D.Float()` ;

### Rectangles:

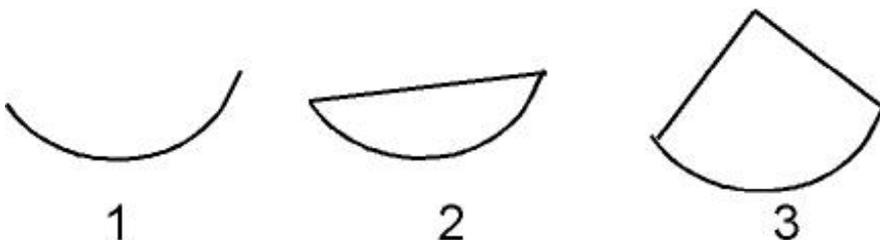
`Rectangle2D.Float rectangle = new Rectangle2D.Float();`

### Ellipses:

`Ellipse2D.Float ellipse = new Ellipse2D.Float();`

### Arcs:

`Arc2D.Float = new Arc2D.Float(.....Arc2D.OPEN);`



- 1) Arc2D.OPEN
- 2) Arc2D.CHORD
- 3) Arc2D.PIE

### Polygones:

- `GeneralPath polly = new GeneralPath(); // création d'un "chemin" encore "vide"`
- `polly.moveTo(5f,0f); // premier trait`
- `polly.moveTo(205f,0f); // second trait`
- `polly.moveTo(5f,90f); // troisième trait`
- `polly.closePath(); // on referme automatiquement par un quatrième "plus court possible" trait`

## Applet Floride2D

Cette applet est une amélioration de Floride présentée au chapitre précédent. Bien sûr, elle fait également partie de l'excellent livre Java2 Plate-Forme de Roger Cadenhead et Laura Lemay aux éditions CampusPress.

```
import java.awt.*;
import java.awt.geom.*;

public class Floride2D extends java.applet.Applet {
    public void paint(Graphics screen) {
        Graphics2D screen2D = (Graphics2D)screen;
        setBackground(Color.blue);

        // dessine les vagues
        screen2D.setColor(Color.white);
        BasicStroke pen = new BasicStroke(2F,
            BasicStroke.CAP_BUTT, BasicStroke.JOIN_ROUND);
        screen2D.setStroke(pen);
        for (int ax = 10; ax < 340; ax += 10)
            for (int ay = 30; ay < 340 ; ay += 10) {
                Arc2D.Float wave = new Arc2D.Float(ax, ay,
                    10, 10, 0, 180, Arc2D.OPEN);
                screen2D.draw(wave);
            }

        // dessine la floride
        GradientPaint gp = new GradientPaint(0F,0F,Color.green,
            50F,50F,Color.orange,true);
        screen2D.setPaint(gp);
        GeneralPath fl = new GeneralPath();
        fl.moveTo(10F,12F);
        fl.lineTo(234F,15F);
        fl.lineTo(253F,25F);
        fl.lineTo(261F,71F);
        fl.lineTo(344F,209F);
        fl.lineTo(336F,278F);
        fl.lineTo(295F,310F);
        fl.lineTo(259F,274F);
        fl.lineTo(205F,188F);
        fl.lineTo(211F,171F);
        fl.lineTo(195F,174F);
        fl.lineTo(191F,118F);
        fl.lineTo(120F,56F);
        fl.lineTo(94F,68F);
        fl.lineTo(81F,49F);
        fl.lineTo(12F,37F);
        fl.closePath();
        screen2D.fill(fl);

        // dessine les ovaes
        screen2D.setColor(Color.black);
        BasicStroke pen2 = new BasicStroke();
        screen2D.setStroke(pen2);
        Ellipse2D.Float e1 = new Ellipse2D.Float(235,140,15,15);
        Ellipse2D.Float e2 = new Ellipse2D.Float(225,130,15,15);
        Ellipse2D.Float e3 = new Ellipse2D.Float(245,130,15,15);
        screen2D.fill(e1);
        screen2D.fill(e2);
        screen2D.fill(e3);
    }
}
```

Ce code HTML est à joindre :

```
<applet code="Floride2D.class" height=370 width=350>  
</applet>
```

Et voilà !

