Exercice G.2: Animation : Différents objets en mouvement

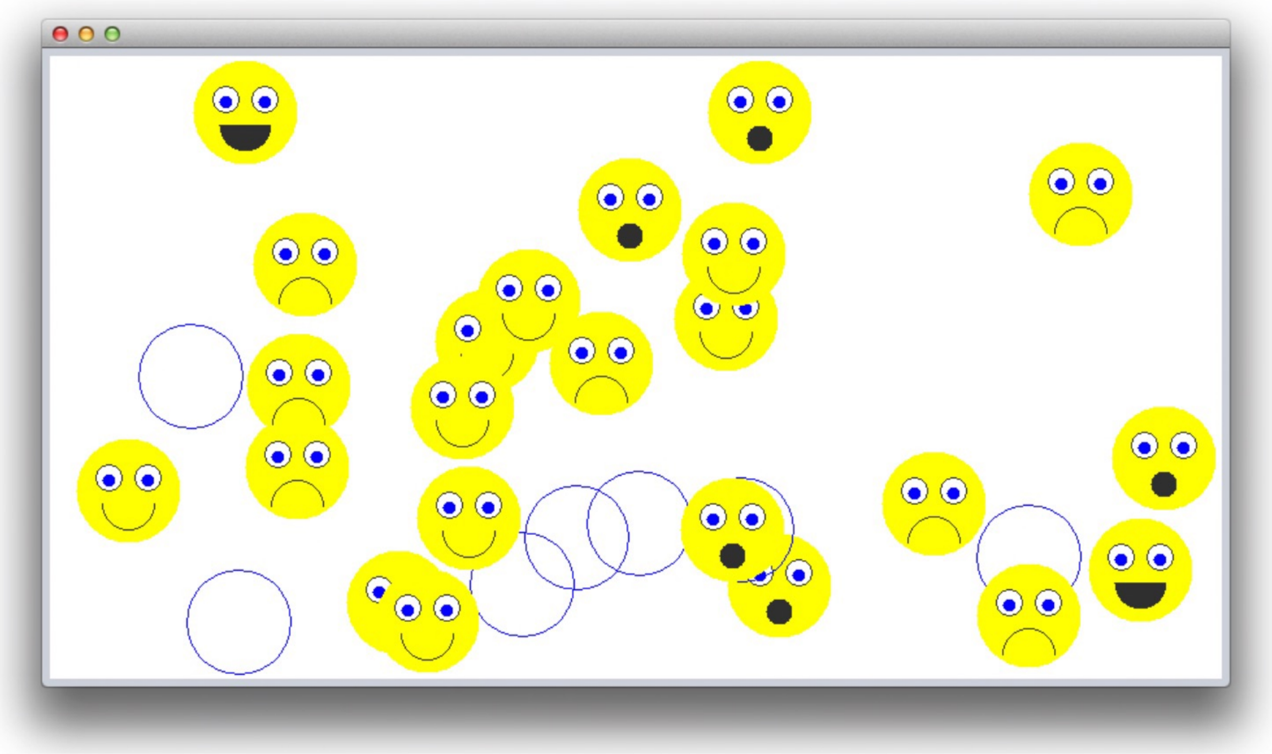
Reprenez l'exercice F3 (*Animation : Boules en mouvement*), puis modifiez-le en suivant les étapes que voici :

Étape 4

Dérivez de **`MovingBall`** les classes **`EmoSmile`**, **`EmoSad`**, **`EmoBigSmile`**, **`EmoSurprised`** qui représentent des *émoticônes* (Smileys) avec différentes expressions. Tous les émoticônes ont la même couleur de fond (jaune) et les mêmes yeux. (Soyez créatifs ! ;-)

Remarque : Consultez l'aide sur les méthodes **`drawArc`** et **`fillArc`**.

Testez votre programme en produisant aléatoirement des balles et différents types d'émoticônes. Profitez au mieux des connaissances que vous avec de l'OOP.



```java

import java.awt.Color;
import java.awt.Graphics;

public class Ball
{
    private int x;
    private int y;
    private int radius;
    private int xStep;
    private int yStep;

    public Ball(int x, int y, int radius, int xStep, int yStep) {
        this.x = x;
        this.y = y;
        this.radius = radius;
        this.xStep = xStep;
        this.yStep = yStep;
    }

    public int getX()
    {
        return x;
    }

    public int getY()
    {
        return y;
    }

    public int getRadius()
    {
        return radius;
    }

    public int getxStep() {
        return xStep;
    }

    public int getyStep() {
        return yStep;
    }

    public void setX(int pX)
    {
        x = pX;
    }

    public void setY(int pY)
    {
        y = pY;
    }

    public void setRadius(int pRadius)
    {
        radius = pRadius;
    }

    public void draw(Graphics g)
    {
        g.setColor(Color.blue);
        g.drawOval(x-radius, y-radius, radius*2+1, radius*2+1);
    }

    public void step(int width, int height)
    {
        if(x+xStep+radius > width) xStep=-xStep;
        if(x+xStep-radius < 0) xStep=-xStep;
        if(y+yStep+radius > height) yStep=-yStep;
        if(y+yStep-radius < 0) yStep=-yStep;
        x=x+xStep;
        y=y+yStep;
    }

    public String toString() {
        return "Ball{" + "x=" + x + ", y=" + y + ", radius=" + radius + ", xStep=" + xStep + ", yStep=" + yStep + '}';
    }

}

```

```java

import java.awt.Graphics;
import java.util.ArrayList;

public class Balls
{
    private ArrayList <Ball> alBalls = new ArrayList<>();
    private ArrayList <Emo> alEmos = new ArrayList<>();

    public void add(Ball ball)
    {
        alBalls.add(ball);
    }

    public Object[] toArray()
    {
        return alBalls.toArray();
    }

    public void draw(Graphics g)
    {
        for(int i=0;i<alBalls.size();i++)
        {
            alBalls.get(i).draw(g);
        }
    }

    public Ball get(int i)
    {
        return alBalls.get(i);
    }

    public void clear()
    {
        alBalls.clear();
    }

    public void step(int width, int height)
    {
        for(int i=0; i<alBalls.size();i++)
        {
            alBalls.get(i).step(width, height);
        }
    }

}

```

```java

import java.awt.Color;
import java.awt.Graphics;

public class DrawPanel extends javax.swing.JPanel {

    private Ball ball;
    private Balls balls;

    public DrawPanel() {
        initComponents();
    }

    public void setBall(Ball pBall)
    {
        ball = pBall;
    }

    public void setBalls(Balls pBalls)
    {
        balls = pBalls;
    }

    public void paintComponent(Graphics g)
    {
        g.setColor(Color.white);
        g.fillRect(0, 0, getWidth(), getHeight());
        if(balls!=null)
        {
            balls.draw(g);
        }
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")

    // Variables declaration - do not modify//GEN-BEGIN:variables
    // End of variables declaration//GEN-END:variables
}

```

```

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public abstract class Emo extends Ball
{
    public Emo(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        g.setColor(Color.yellow);
        g.fillOval(super.getX()-super.getRadius(), super.getY()-super.
getRadius(), super.getRadius()*2+1, super.getRadius()*2+1);
        g.setColor(Color.white);
        g.fillOval(super.getX()-super.getRadius()*2/3, super.getY()-super.
getRadius()/3*2, super.getRadius()*2/5, super.getRadius()*2/5);
        g.fillOval(super.getX()+super.getRadius()*3/10, super.getY()-super.
getRadius()/3*2, super.getRadius()*2/5, super.getRadius()*2/5);
        g.setColor(Color.black);
        g.drawOval(super.getX()-super.getRadius()*2/3+1, super.getY()-super.
getRadius()/3*2, super.getRadius()*2/5+1, super.getRadius()*2/5+1);
        g.drawOval(super.getX()+super.getRadius()*3/10+1, super.getY()-
super.getRadius()/3*2, super.getRadius()*2/5+1, super.getRadius()*2/5+1);
        g.setColor(Color.blue);
        g.fillOval(super.getX()-super.getRadius()/3*5/3, super.getY()-super.
getRadius()*3/5, super.getRadius()*2/8, super.getRadius()*2/8);
        g.fillOval(super.getX()+super.getRadius()*4/10, super.getY()-super.
getRadius()*3/5, super.getRadius()*2/8, super.getRadius()*2/8);
    }
}

```

```java

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public class EmoBigSmile extends Emo
{
    public EmoBigSmile(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        super.draw(g);
        g.setColor(Color.black);
        g.fillArc(super.getX()-super.getRadius()/2,super.getY()+super.getRadius()/40, super.getRadius(),super.getRadius()*2/3,0,-180);
    }
}

```

```java

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public class EmoNaughty extends Emo
{
    private boolean isTongueOut = false;

    public EmoNaughty(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        super.draw(g);
        g.setColor(Color.black);
        g.drawLine(super.getX()-super.getRadius()/2, super.getY()+super.
getRadius()/3,super.getX()+super.getRadius()/2, super.getY()+super.
getRadius()/3);
    }
}

```

```java

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public class EmoSad extends Emo
{
    public EmoSad(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        super.draw(g);
        g.setColor(Color.black);
        g.drawArc(super.getX()-super.getRadius()/3*5/3,super.getY()+super.getRadius()/20, super.getRadius(),super.getRadius()*2/3,0,180);
    }
}

```

```java

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public class EmoSmile extends Emo
{
    public EmoSmile(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        super.draw(g);
        g.setColor(Color.black);
        g.drawArc(super.getX()-super.getRadius()/3*5/3,super.getY()+super.getRadius()/20, super.getRadius(),super.getRadius()*2/3,0,-180);
    }
}

```

```java

import java.awt.Color;
import java.awt.Graphics;

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author luxformel
 */
public class EmoSurprised extends Emo
{
    public EmoSurprised(int x, int y, int radius, int xStep, int yStep)
    {
        super(x, y, radius, xStep, yStep);
    }

    public void draw(Graphics g)
    {
        super.draw(g);
        g.setColor(Color.black);
        g.fillOval(super.getX()-super.getRadius()/5, super.getY()+super.
getRadius()/5, super.getRadius()/2, super.getRadius()/2);
    }
}

```

```java

import javax.swing.Timer;

public class MainFrame extends javax.swing.JFrame {

    Ball ball = null;
    Balls balls = new Balls();
    private int xDirection=1;
    private int step=5;
    private int speed=10;
    private int yDirection=1;
        Timer timer;

    public MainFrame() {
        initComponents();
        drawPanel.setBalls(balls);
        drawPanel.setBall(ball);
        timer= new Timer(speed, stepB.getActionListeners()[0]);
        ballList.setListData(balls.toArray());
        updateView();
    }

    public void updateView()
    {
        ballList.setListData(balls.toArray());
        drawPanel.repaint();
    }

    @SuppressWarnings("unchecked")

    private void stepBActionPerformed(java.awt.event.ActionEvent evt) {
        balls.step(drawPanel.getWidth(), drawPanel.getHeight());
        drawPanel.repaint();
        updateView();
    }

    private void startStopBActionPerformed(java.awt.event.ActionEvent evt)
{
        if(startStopB.getText().equals("Start"))
        {
            timer.start();
            startStopB.setText("Stop");
        }
        else
        {
            timer.stop();
            startStopB.setText("Start");
        }
        updateView();
    }

    private void addBActionPerformed(java.awt.event.ActionEvent evt) {
        int radius = 50;//(int)(Math.random()*(50-20+1))+20;
        int x = (int)(Math.random()*((drawPanel.getWidth()-radius)-0+1))+0;
        int y = (int)(Math.random()*((drawPanel.getWidth()-radius)-0+1))+0;
        while(x+radius*2>=drawPanel.getWidth() || y+radius*2>=drawPanel.
getHeight() ||x-radius*2<=0 || y-radius*2<=0)
        {
            x = (int)(Math.random()*((drawPanel.getWidth()-radius)-0+1))+0;
            y = (int)(Math.random()*((drawPanel.getWidth()-radius)-0+1))+0;
        }
        int xStep =(int)(Math.random()*(8-1+1))+1;
        int yStep =(int)(Math.random()*(8-1+1))+1;
        int random = (int)(Math.random()*(5-1+1))+1;
        if(random==1)
        {
            balls.add(new Ball(x,y,radius,xStep,yStep));
        }
        else if(random==2)
        {
            balls.add(new EmoSmile(x,y,radius,xStep,yStep));
        }
        else if(random==3)
        {
            balls.add(new EmoSad(x,y,radius,xStep,yStep));
        }
        else if(random==4)
        {
            balls.add(new EmoBigSmile(x,y,radius,xStep,yStep));
        }
        else
        {
            balls.add(new EmoSurprised(x,y,radius,xStep,yStep));
        }
        //balls.add(new EmoNaughty(x,y,radius,xStep,yStep));
        updateView();
    }

    private void clearBActionPerformed(java.awt.event.ActionEvent evt) {
        balls.clear();
        updateView();
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new MainFrame().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton addB;
    private javax.swing.JList ballList;
    private javax.swing.JButton clearB;
    private DrawPanel drawPanel;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JButton startStopB;
    private javax.swing.JButton stepB;
    // End of variables declaration//GEN-END:variables
}

```