

```
import java.awt.Color;
import java.awt.Graphics;
public class Ball
{
    private int x = 30;
    private int y = 30;
    private int radius = 30;

    public int getX()
    {
        return x;
    }

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

    public int getY()
    {
        return y;
    }

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

    public int getRadius()
    {
        return radius;
    }

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

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

```
import java.awt.Color;
import java.awt.Graphics;
public class DrawPanel extends javax.swing.JPanel
{
    private Ball ball = null;

    public DrawPanel()
    {
        initComponents();
    }

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

    public void paintComponent(Graphics g)
    {
        // clean the background
        g.setColor(Color.WHITE);
        g.fillRect(0, 0, getWidth(), getHeight());

        // draw the ball
        if (ball != null)
            ball.draw(g);
    }
// Skipped: ... initComponents { ... }
// Variables declaration - do not modify//GEN-BEGIN:variables
// End of variables declaration//GEN-END:variables
}
```

```

import javax.swing.Timer;
public class MainFrame extends javax.swing.JFrame
{
    private Ball ball = new Ball();
    private int xDir = 1; // horizontal direction of the ball (+1 -> right / -1 -> left)
    private int yDir = 1; // vertical direction of the ball (+1 -> down / -1 -> up)
    private int step = 5;
    private int delay = 100;
    private Timer timer;

    public MainFrame()
    {
        initComponents();
        drawPanel.setBall(ball);
        timer = new Timer(delay, stepButton.getActionListeners()[0]);
        stepButton.setVisible(false);
    }
// Skipped: ... initComponents { ... }
    private void startStopButtonActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_startStopButtonActionPerformed
        if (timer.isRunning())
        {
            timer.stop();
            startStopButton.setText("Start");
        }
        else
        {
            timer.start();
            startStopButton.setText("Stop");
        }
    } //GEN-LAST:event_startStopButtonActionPerformed

    private void stepButtonActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_stepButtonActionPerformed
        int x = ball.getX();
        int y = ball.getY();
        int r = ball.getRadius();

        // Solution plus logique - la direction positive (+1) ou négative (-1)
        // est prise en compte au départ
        //if (((xDir > 0) && (x + r + step > drawPanel.getWidth())) ||
        //    ((xDir < 0) && (x - r - step < 0)))
        //{
        //    xDir = -xDir;
        //}
        //if (((yDir > 0) && (y + r + step > drawPanel.getHeight())) ||
        //    ((yDir < 0) && (y - r - step < 0)))
        //{
        //    yDir = -yDir;
        //}

        // Solution plus "rapide" en combinant la direction avec le pas
        if ((x + r + step * xDir > drawPanel.getWidth()) ||
            (x - r + step * xDir < 0))
        {
            xDir = -xDir;
        }
        ball.setX(x + step * xDir);

        if ((y + r + step * yDir > drawPanel.getHeight()) ||
            (y - r + step * yDir < 0))
        {
            yDir = -yDir;
        }
        ball.setY(y + step * yDir);

        repaint();
    } //GEN-LAST:event_stepButtonActionPerformed
// Skipped: ... Look & Feel
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private DrawPanel drawPanel;
    private javax.swing.JButton startStopButton;
    private javax.swing.JButton stepButton;
    // End of variables declaration//GEN-END:variables
}

```