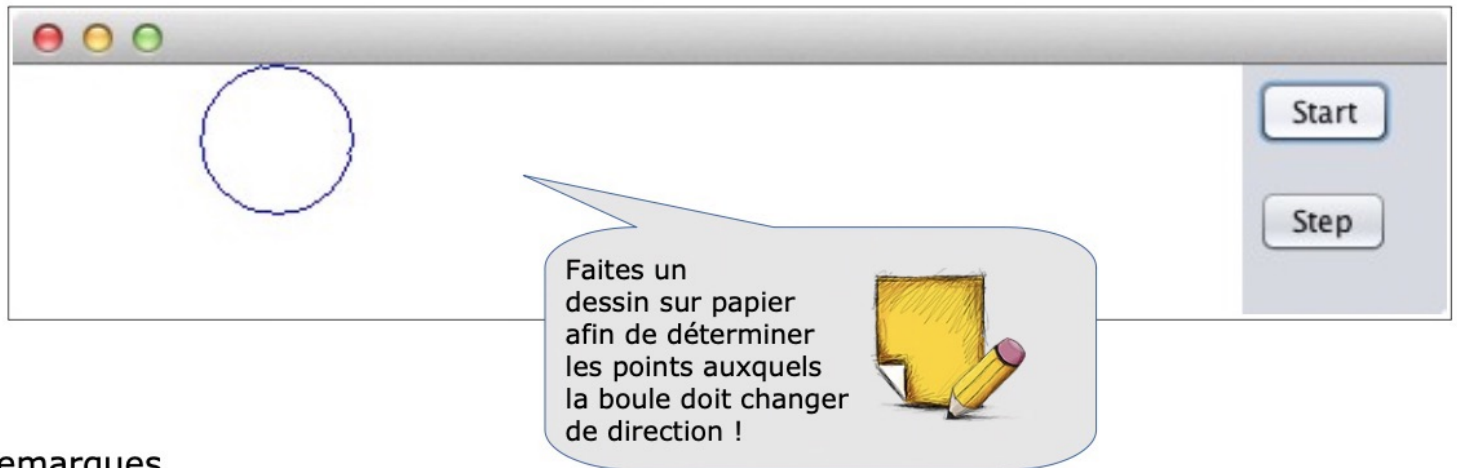


Étape 1

Écrivez d'abord un programme qui fait bouger une boule du côté gauche vers le côté droit et vice-versa.



Remarques

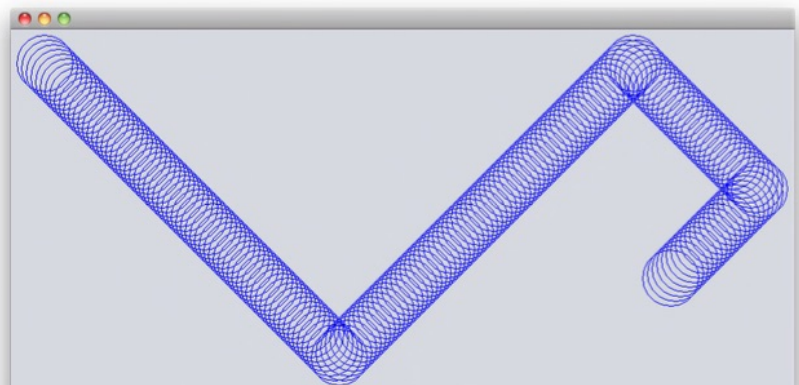
- Pour tester, le bouton **stepButton** permet de déplacer la boule d'une étape lorsque le chronomètre n'est pas encore actif. Ceci nous permet de vérifier exactement les mouvements de la boule. Plus tard, il peut être rendu invisible à l'aide de la méthode **setVisible(boolean)**.
- L'animation peut être lancée et arrêtée par le bouton **startStopButton**.
- La boule peut être représentée par la classe suivante, (x,y) étant les coordonnées du **centre** de la boule :

Ball
- x : int
- y : int
- radius : int
+ getX() : int
+ setX(pX : int) : void
+ getY() : int
+ setY(pY : int) : void
+ getRadius() : int
+ setRadius(pRadius : int) : void
+ draw(g : Graphics) : void

Étape 2

Modifiez votre programme de manière à ce que la boule ne fasse pas uniquement un mouvement horizontal mais aussi un mouvement vertical !

Voici une capture d'écran truquée afin de vous permettre de mieux saisir le principe de fonctionnement de la modification à apporter au programme.



Question à discuter:

Que peut-on dire de ce programme du point de vue encapsulation et modélisation MVC?

```
1
2 import java.awt.Color;
3 import java.awt.Graphics;
4
5 /*
6  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.
7  * txt to change this license
8  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
9  * this template
10 */
11
12 /**
13  * @author luxformel
14  */
15 public class Ball {
16     private int x;
17     private int y;
18     private int radius;
19
20     public int getX() {
21         return x;
22     }
23
24     public void setX(int x) {
25         this.x = x;
26     }
27
28     public int getY() {
29         return y;
30     }
31
32     public void setY(int y) {
33         this.y = y;
34     }
35
36     public int getRadius() {
37         return radius;
38     }
39
40     public void setRadius(int radius) {
41         this.radius = radius;
42     }
43
44     public void draw(Graphics g){
45         g.setColor(Color.black);
46         g.drawOval(x - radius, y - radius, 2 * radius, 2 * radius);
47     }
48 }
```

```
1
2 import java.awt.Color;
3 import java.awt.Graphics;
4
5 /*
6  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.
7  * txt to change this license
8  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JPanel.java to
9  * edit this template
10 */
11
12 /**
13  * @author luxformel
14  */
15 public class DrawPanel extends javax.swing.JPanel {
16     private Ball ball = null;
17
18     public void setBall(Ball ball) {
19         this.ball = ball;
20     }
21
22     /**
23      * Creates new form DrawPanel
24      */
25     public DrawPanel() {
26         initComponents();
27     }
28
29     @Override
30     protected void paintComponent(Graphics g) {
31         super.paintComponent(g); // Generated from nbfs:
32         //nbhost/SystemFileSystem/Templates/Classes/Code/OverriddenMethodBody
33         g.setColor(Color.white);
34         g.fillRect(0, 0, getWidth(), getHeight());
35         if (ball != null){
36             ball.draw(g);
37         }
38     }
39
40
41     /**
42      * This method is called from within the constructor to initialize the
43      * form.
44      * WARNING: Do NOT modify this code. The content of this method is
45      * always
46      * regenerated by the Form Editor.
47      */
48     @SuppressWarnings("unchecked")
49
50 }
```

```
61
62
63     // Variables declaration – do not modify//GEN-BEGIN:variables
64     // End of variables declaration//GEN-END:variables
65 }
66
```

```
1
2  import javax.swing.Timer;
3
4  /*
4   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.
5   txt to change this license
5   * Click nbfs://nbhost/SystemFileSystem/Templates/GuiForms/JFrame.java to
6   edit this template
7   */
8
9  /**
10   *
11   * @author luxformel
12   */
13  public class MainFrame extends javax.swing.JFrame {
14      private Ball ball = new Ball();
15      private int step = 5;
16      private Timer timer;
17
18      /**
19       * Creates new form MainFrame
20       */
21      public MainFrame() {
22          initComponents();
23          ball.setX(100);
24          ball.setY(100);
25          ball.setRadius(25);
26          drawPanel.setBall(ball);
27          drawPanel.repaint();
28          timer = new Timer(100, stepButton.getActionListeners()[0]);
29      }
30
31      /**
31       * This method is called from within the constructor to initialize the
32       form.
32       * WARNING: Do NOT modify this code. The content of this method is
33       always
34       * regenerated by the Form Editor.
35       */
38      @SuppressWarnings("unchecked")
42
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
100  private void stepButtonActionPerformed(java.awt.event.ActionEvent evt)
101  {
102
103      int oldX = ball.getX();
```

```
104
105     if (ball.getX() + ball.getRadius() + step > drawPanel.getWidth()){
106         ball.setX(drawPanel.getWidth() - ball.getRadius());
107         step = -5;
108     }
109     else{
110         ball.setX(oldX + step);
111     }
112
113     if (ball.getX() - ball.getRadius() + step < 0){
114         ball.setX(ball.getRadius());
115         step = 5;
116     }
117     else{
118         ball.setX(oldX + step);
119     }
120
121     drawPanel.repaint();
122 }
123
123 private void startStopButtonActionPerformed(java.awt.event.ActionEvent
124 evt) {
125     if (timer.isRunning()){
126         timer.stop();
127         startStopButton.setText("Start");
128     }
129     else{
130         timer.start();
131         startStopButton.setText("Stop");
132     }
133 }
134
135 /**
136  * @param args the command line arguments
137  */
138 public static void main(String args[]) {
139     /* Set the Nimbus look and feel */
140
141     /* Create and display the form */
142     java.awt.EventQueue.invokeLater(new Runnable() {
143         public void run() {
144             new MainFrame().setVisible(true);
145         }
146     });
147 }
148
149 // Variables declaration - do not modify//GEN-BEGIN:variables
150 private DrawPanel drawPanel;
151 private javax.swing.JButton startStopButton;
152 private javax.swing.JButton stepButton;
153 // End of variables declaration//GEN-END:variables
154 }
```

