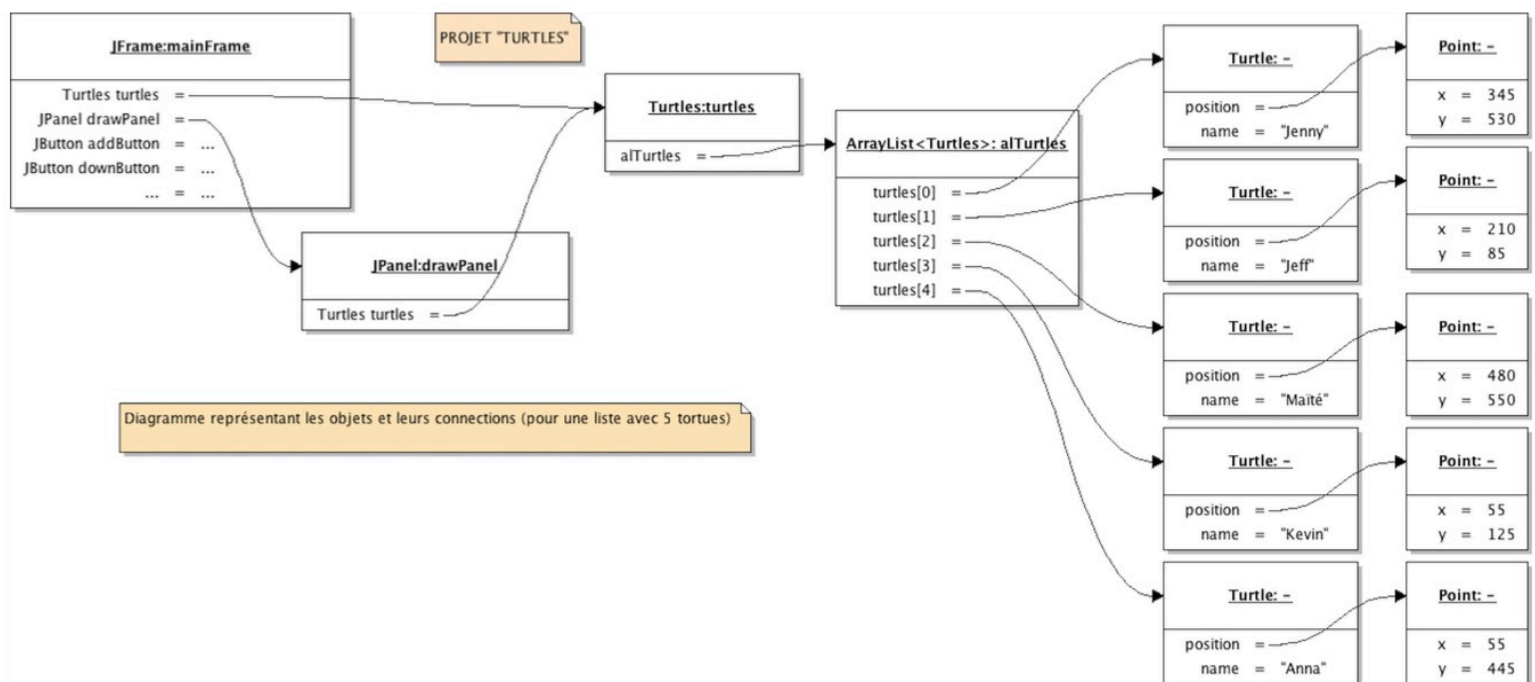


Exercice 3:

Tortues en couleur

Ajoutez un attribut **color** à la classe **Turtle**, qui est initialisé lors de la création d'une tortue. Lors du dessin, la tortue est dessinée avec la couleur **color**. Ajoutez à l'interface graphique une possibilité de choisir une couleur.

Conseil : Ajoutez un composant **JColorChooser** à votre fiche et nommez-le **colorChooser**. Ainsi vous avez accès à sa couleur actuelle par : **colorChooser.getColor()**



```
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 Turtles turtles;
17
18     public void setTurtles(Turtles t) {
19         this.turtles = t;
20     }
21
22     /**
23      * Creates new form DrawPanel
24      */
25     public DrawPanel() {
26         initComponents();
27     }
28
29     /**
30      * This method is called from within the constructor to initialize the
31      * form.
32      * WARNING: Do NOT modify this code. The content of this method is
33      * always
34      * regenerated by the Form Editor.
35      */
36     @SuppressWarnings("unchecked")
37
38
39
40
41     @Override
42     protected void paintComponent(Graphics g) {
43         super.paintComponent(g); // Generated from nbfs:
44         //nbhost/SystemFileSystem/Templates/Classes/Code/OverriddenMethodBody
45         //Dessin d'un fond blanc sur le drawPanel
46         g.setColor(Color.white);
47         g.fillRect(0, 0, getWidth(), getHeight());
48
49         //Ce contrôle doit toujours être effectué, afin d'éviter des erreurs
50         //genre "Null pointer exception" directement dans le MainFrame
51         if (turtles != null)
52             turtles.draw(g);
53     }
54 }
```

```
1
2 import java.awt.Color;
3 import java.awt.Point;
4 import javax.swing.JColorChooser;
5
6 /*
6  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.
7  * txt to change this license
7  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
8  * edit this template
9  */
10
11 /**
12  *
13  * @author luxformel
14  */
15 public class MainFrame extends javax.swing.JFrame {
16
17     //Instanciación / Création d'un nouvel objet de type Turtles
18     private Turtles turtles = new Turtles();
19     private Color color = Color.black;
20     /**
21      * Creates new form MainFrame
22      */
23     public MainFrame() {
24         initComponents();
25
26         //L'objet instancié de type turtles doit être passé au drawPanel
27         drawPanel.setTurtles(turtles);
28
29         //Raffraîchir l'interface
30         updateView();
31         //positionLabel.setText("X:" + position.x + "; Y:" + position.y);
32     }
33
34     public void updateView(){
35         //Redessine la surface de dessin
36         drawPanel.repaint();
37
38         //Affichage de la ArrayList contenant les tortues dans la JList
39         (Vitrine) turtlesList.setListData(turtles.toArray());
40     }
41
42     /**
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")
```

```
62
64
75
82
89
96
103
105
107
114
121
127
134
207
210
211
211 private void rightButtonActionPerformed(java.awt.event.ActionEvent evt)
212 {
213     //Récupérer l'indice de la tortue sélectionnée
214     int index = turtlesList.getSelectedIndex();
215     //Si une tortue est sélectionnée
215     //Alors la tortue sélectionnée de la ArrayList est mise dans la
216 variable t
217     if (index != -1){
218         //Récupère la tortue à l'indice index
219         Turtle t = turtles.get(index);
220
221         //Déplace la tortue
222         t.goRight(10, drawPanel.getWidth());
223
224
225         updateView();
226
227         //Resélectionne la tortue dans la JList
228         turtlesList.setSelectedIndex(index);
229     }
230 }
231
231 private void leftButtonActionPerformed(java.awt.event.ActionEvent evt)
232 {
233     //Récupérer l'indice de la tortue sélectionnée
234     int index = turtlesList.getSelectedIndex();
235     //Si une tortue est sélectionnée
235     //Alors la tortue sélectionnée de la ArrayList est mise dans la
236 variable t
237     if (index != -1){
238         Turtle t = turtles.get(index);
239         t.goLeft(10);
240         updateView();
241         turtlesList.setSelectedIndex(index);
242     }
243 }
```

```
244
244     private void downButtonActionPerformed(java.awt.event.ActionEvent evt)
245 {
246     //Récupérer l'indice de la tortue sélectionnée
247     int index = turtlesList.getSelectedIndex();
248     //Si une tortue est sélectionnée
248     //Alors la tortue sélectionnée de la ArrayList est mise dans la
249 variable t
250     if (index != -1){
251         Turtle t = turtles.get(index);
252         t.goDown(10, drawPanel.getHeight());
253         updateView();
254         turtlesList.setSelectedIndex(index);
255     }
256 }
257
258     private void upButtonActionPerformed(java.awt.event.ActionEvent evt) {
259         //Récupérer l'indice de la tortue sélectionnée
260         int index = turtlesList.getSelectedIndex();
261         //Si une tortue est sélectionnée
261         //Alors la tortue sélectionnée de la ArrayList est mise dans la
262 variable t
263         if (index != -1){
264             Turtle t = turtles.get(index);
265             t.goUp(10);
266
267             updateView();
268
269             turtlesList.setSelectedIndex(index);
270         }
271     }
272 }
273
273     private void addTurtleButtonActionPerformed(java.awt.event.ActionEvent
274 evt) {
275         int minX = 10;
276         int maxX = drawPanel.getWidth() - 10;
277         int minY = 13;
278         int maxY = drawPanel.getHeight() - 10;
279
280
281         int x = (int)(Math.random() * (maxX - minX + 1)) + minX;
282         int y = (int)(Math.random() * (maxY - minY + 1)) + minY;
283
284         Point position = new Point (x, y);
285         String name = nameTextField.getText();
286
287         //if (! name.isEmpty()){
288         //Seulement si le champ de texte n'est pas vide
289         //alors une tortue est instanciée et ajoutée
290         if (! nameTextField.getText().equals("")){
291             //Instanciation de la tortue
```

```

292         Turtle t = new Turtle(position, name, color);
293
294         //Ajout de la tortue à la liste contenue dans turtles
295         turtles.add(t);
296         updateView();
297     }
298 }
299
300 private void nameTextFieldActionPerformed(java.awt.event.ActionEvent
301 evt) {
302     // TODO add your handling code here:
303 }
304
305 private void colorButtonActionPerformed(java.awt.event.ActionEvent evt)
306 {
307     //Choisir la couleur de la tortue dans un JColorChooser.
308     //L'attribut color est passé en paramètre de la méthode showDialog,
309     //afin d'avoir dans la fenêtre de dialogue la même couleur qui a
310     déjà été définie
311     color = JColorChooser.showDialog(rootPane, "Please choose a color
312 for the turtle", color);
313 }
314
315 /**
316  * @param args the command line arguments
317  */
318 public static void main(String args[]) {
319     /* Set the Nimbus look and feel */
320
321     /* Create and display the form */
322     java.awt.EventQueue.invokeLater(new Runnable() {
323         public void run() {
324             new MainFrame().setVisible(true);
325         }
326     });
327 }
328
329 // Variables declaration - do not modify//GEN-BEGIN:variables
330 private javax.swing.JButton addTurtleButton;
331 private javax.swing.JButton colorButton;
332 private javax.swing.JButton downButton;
333 private DrawPanel drawPanel;
334 private javax.swing.JLabel jLabel1;
335 private javax.swing.JScrollPane jScrollPane1;
336 private javax.swing.JButton leftButton;
337 private javax.swing.JTextField nameTextField;
338 private javax.swing.JLabel positionLabel;
339 private javax.swing.JButton rightButton;
340 private javax.swing.JList turtlesList;
341 private javax.swing.JButton upButton;
342 // End of variables declaration//GEN-END:variables
343 }

```

```
1
2 import java.awt.Color;
3 import java.awt.Graphics;
4 import java.awt.Point;
5
6
7 /*
7  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.
8 txt to change this license
8  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
9 this template
10 */
11
12 /**
13  *
14  * @author luxformel
15  */
16 public class Turtle {
17     //Déclaration des attributs
18     private Point position;
19     private String name;
20     private Color color;
21
22     //Déclaration et définition du constructeur
23     public Turtle(Point position, String pName, Color pColor) {
24         this.position = position;
25         name = pName;
26         color = pColor;
27     }
28
29     //Getters & setters
30     public Point getPosition() {
31         return position;
32     }
33
34
35     public void goRight(int pDist, int width){
36         if (position.x + 10 + pDist <= width)
37             position.x = position.x + pDist;
38         else
39             position.x = width - 10;
40     }
41
42     public void goLeft(int pDist){
43         if (position.x - 10 - pDist >= 0)
44             position.x = position.x - pDist;
45         else
46             position.x = 10;
47     }
48
49     public void goUp(int pDist){
50         if (position.y - 13 - pDist >= 0)
```

```
51         position.y = position.y - pDist;
52     else
53         position.y = 13;
54 }
55
56 public void goDown(int pDist, int height){
57     if (position.y + 10 + pDist <= height)
58         position.y = position.y + pDist;
59     else
60         position.y = height - 10;
61 }
62
63 //Cette méthode dessine la tortue
64 public void draw(Graphics g){
65     g.setColor(color);
66     g.fillOval(position.x - 7, position.y - 7, 15, 15);
67     g.fillOval(position.x - 2, position.y - 13, 5, 6);
67     g.drawLine(position.x - 10, position.y - 7, position.x + 10,
68 position.y + 10);
68     g.drawLine(position.x - 10, position.y + 10, position.x + 10,
69 position.y - 7);
70 }
71
72 public String toString(){
73     return name + " (X:" + position.x + ";" + " Y:" + position.y + ")";
74 }
75
76 }
77
```



```
1
2 import java.awt.Graphics;
3 import java.util.ArrayList;
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 Turtles {
16     //Liste contenant toutes les tortues
17     private ArrayList<Turtle> alTurtles = new ArrayList<>();
18
19     public Object[] toArray() {
20         return alTurtles.toArray();
21     }
22
23     public Turtle get(int index) {
24         return alTurtles.get(index);
25     }
26
27     public boolean add(Turtle e) {
28         return alTurtles.add(e);
29     }
30
31     public void draw(Graphics g){
32         for(int i = 0; i < alTurtles.size(); i++){
33             alTurtles.get(i).draw(g);
34         }
35     }
36 }
```