Laboratoire 1c

MCR: Laboratoire

24-25 / SRS

Objectif

• Utiliser les modèles : singleton, fabrication et fabrique abstraite.

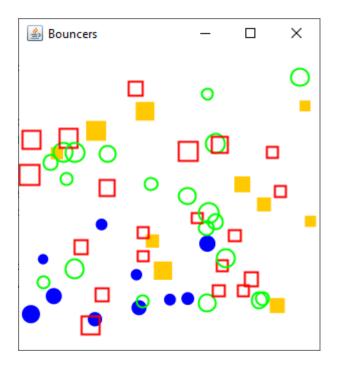
Donnée

Dernière étape de ce laboratoire, il faut compléter ce qui a été fait précédemment.

En plus des cercles et carrés pleins, il faut pouvoir créer et afficher une version avec une bordure (de 2 pixels, voir BasicStroke d'AWT).

Il y a donc 2 familles de formes :

- Pleine : cercles affichés en bleu, carrés affichés en orange
- Avec une bordure : cercles affichés en vert, carrés affichés en rouge



Il faut gérer (en utilisant un KeyAdapter, l'interface Displayer a été complétée en conséquence) l'appui sur les touches du clavier suivantes :

- e : efface les formes
- f: génère 10 cercles et 10 carrées pleins
- b : génère 10 cercles et 10 carrés avec une bordure
- q : quitte le programme

La fenêtre doit pourvoir être redimensionnée et son affichage initial doit être vide.

HEIG-VD MCR: Laboratoire Département TIC 24-25 / SRS

Les formes doivent implémenter l'interface Bouncable et leur affichage (déclenché par l'appel à draw()) doit être délégué à un objet implémentant l'interface Renderer. La liste de formes dans BouncersApp doit donc être de type Bouncable (voir annexes).

Pour l'affichage dans la fenêtre graphique, il peut être utile de passer par une Image obtenue en appelant la méthode createImage (disponible sur un JPanel) afin d'utiliser son Graphics2D (obtenu par la méthode getGraphics). Ensuite les formes peuvent être dessinées sur cet objet, qui pourra être réaffiché sur le JPanel en utilisant la méthode drawImage (du Graphics2D du JPanel). La méthode fill (appelée sur un Graphics2D) peut aussi être utile.

A réaliser

Par groupe de 2

Travail à rendre le 19.03.2025 à 13h15

A remettre, sur Cyberlearn

- Le diagramme de classes
- Le code source

Annexes

}

```
Note: les classes Graphics2D, KeyAdapter, Color et Shape sont celles d'AWT.
public interface Displayer {
  int getWidth();
  int getHeight();
  Graphics2D getGraphics();
  void repaint();
 void setTitle(String s);
 void addKeyListener(KeyAdapter ka);
}
public interface Bouncable {
  void draw();
  void move();
 Color getColor();
 Shape getShape();
public interface Renderer {
 void display(Graphics2D g, Bouncable b);
public class BouncersApp {
  private final LinkedList<Bouncable> bouncers = new LinkedList<>();
  /* ... */
  public BouncersApp() { /* ... */ }
  public void run() { /* ... */ }
  public static void main(String... args) {
    new BouncersApp().run();
  }
```