

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**Лабораторна робота №** **3**

з дисципліни “Математичні та алгоритмічні основи комп’ютерної графіки”

|  |  |  |  |
| --- | --- | --- | --- |
| Виконав  студент III курсу  групи КП-83  Мар’яненко Роман Геннадійович  (*прізвище, ім’я, по батькові*)  варіант № 11 |  |  | Зарахована  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладачем  Шкурат Оксаною Сергіївною (*прізвище, ім’я, по батькові*) |

Київ 2020

**Варіант завдання**

**Завдання:** За допомогою примітивів JavaFX максимально реально зобразити персонажа за варіантом та виконати його 2D анімацію. Для анімації скористатися стандартними засобами бібліотеки JavaFX. Обов’язковою є реалізація таких видів анімації:

1. Переміщення;
2. Поворот;
3. Масштабування.

**Варіант:**



**Лістинг коду програми**

|  |
| --- |
| Main.java |
| package lab3;  import javafx.animation.\*;  import javafx.application.Application;  import javafx.scene.Group;  import javafx.scene.Scene;  import javafx.scene.paint.Color;  import javafx.scene.paint.Paint;  import javafx.scene.shape.\*;  import javafx.scene.transform.Rotate;  import javafx.stage.Stage;  import javafx.util.Duration;  public class Main extends Application {  Paint bgColor = Color.rgb(238, 255, 138);  Group root;  Scene scene;  public static void main (String args[]) {  launch(args);  }  @Override  public void start(Stage primaryStage) {  Group root = new Group();  Scene scene = new Scene (root, 600, 500);  scene.setFill(bgColor);  // Big mushroom  root.getChildren().add(BigMushroom.getLowerPartOfStipe());  root.getChildren().add(BigMushroom.getUpperPartOfStipe());  root.getChildren().add(BigMushroom.getCap());  root.getChildren().addAll(BigMushroom.getCapStains());  // Small mushroom  root.getChildren().add(SmallMushroom.getStipe());  root.getChildren().add(SmallMushroom.getCap());  root.getChildren().addAll(SmallMushroom.getCapStains());  // Animation  int duration = 3000;  ScaleTransition scaleFrom = new ScaleTransition(Duration.millis(duration), root);  scaleFrom.setToX(1);  scaleFrom.setToY(1);  ScaleTransition scaleTo = new ScaleTransition(Duration.millis(duration), root);  scaleTo.setToX(0.1);  scaleTo.setToY(0.1);  RotateTransition rotate = new RotateTransition(Duration.millis(duration), root);  rotate.setByAngle(360f);  rotate.setCycleCount(Timeline.INDEFINITE);  TranslateTransition translateTo = new TranslateTransition(Duration.millis(duration \* 2), root);  translateTo.setFromX(0);  translateTo.setToX(300);  translateTo.setCycleCount(Timeline.INDEFINITE);  translateTo.setAutoReverse(true);  TranslateTransition translateFrom = new TranslateTransition(Duration.millis(duration \* 2), root);  translateFrom.setFromX(300);  translateFrom.setToX(0);  translateFrom.setCycleCount(Timeline.INDEFINITE);  translateFrom.setAutoReverse(true);  SequentialTransition scale = new SequentialTransition();  scale.getChildren().addAll(  scaleTo,  scaleFrom  );  scale.setCycleCount(Timeline.INDEFINITE);  SequentialTransition translate = new SequentialTransition();  translate.getChildren().addAll(  translateTo,  translateFrom  );  translate.setCycleCount(Timeline.INDEFINITE);  ParallelTransition animations = new ParallelTransition();  animations.getChildren().addAll(  scale,  rotate,  translate  );  animations.play();  // End of animation  primaryStage.setTitle("Mushrooms");  primaryStage.setScene(scene);  primaryStage.show();  }  } |

|  |
| --- |
| SmallMushroom.java |
| package lab3;  import javafx.scene.paint.Color;  import javafx.scene.paint.Paint;  import javafx.scene.shape.Circle;  import javafx.scene.shape.Polygon;  import javafx.scene.shape.StrokeType;  public class SmallMushroom {  static Paint capColor = Color.rgb(185, 13, 11);  static Paint stipeColor = Color.rgb(245, 247, 196);  static public Polygon getStipe() {  Polygon stipe = new Polygon(  240.0, 450.0,  250.0, 445.0,  260.0, 435.0,  260.0, 420.0,  250.0, 400.0,  245.0, 355.0,  218.0, 355.0,  217.0, 400.0,  210.0, 420.0,  220.0, 440.0  );  stipe.setFill(stipeColor);  stipe.setStroke(Color.BLACK);  stipe.setStrokeType(StrokeType.OUTSIDE);  return stipe;  }  static public Polygon getCap() {  Polygon cap = new Polygon(  245.0, 355.0,  255.0, 345.0,  265.0, 330.0,  265.0, 315.0,  255.0, 300.0,  240.0, 290.0,  220.0, 288.0,  210.0, 295.0,  200.0, 305.0,  195.0, 315.0,  195.0, 335.0,  201.0, 352.0,  218.0, 360.0  );  cap.setFill(capColor);  cap.setStroke(Color.BLACK);  cap.setStrokeType(StrokeType.OUTSIDE);  return cap;  }  static public Circle[] getCapStains() {  Circle[] stains = new Circle[7];  stains[0] = getStain(232.0, 340.0, 5);  stains[1] = getStain(247.0, 322.0, 2);  stains[2] = getStain(227.0, 320.0, 5);  stains[3] = getStain(210.0, 330.0, 4);  stains[4] = getStain(245.0, 300.0, 3);  stains[5] = getStain(225.0, 300.0, 4);  stains[6] = getStain(250.0, 332.0, 3);  return stains;  }  static private Circle getStain(double x, double y, double r) {  Circle stain = new Circle(x, y, r);  stain.setFill(stipeColor);  stain.setStroke(Color.BLACK);  stain.setStrokeType(StrokeType.OUTSIDE);  return stain;  }  } |

|  |
| --- |
| BigMushroom.java |
| package lab3;  import javafx.scene.paint.Color;  import javafx.scene.paint.Paint;  import javafx.scene.shape.Circle;  import javafx.scene.shape.Ellipse;  import javafx.scene.shape.Polygon;  import javafx.scene.shape.StrokeType;  public class BigMushroom {  static Paint capColorPrimary = Color.rgb(185, 13, 11);  static Paint capColorSecondary = Color.rgb(247, 125, 120);  static Paint stipeColor = Color.rgb(245, 247, 196);  static public Polygon getLowerPartOfStipe() {  Polygon stipe = new Polygon(  110.0, 420.0,  120.0, 435.0,  140.0, 440.0,  155.0, 435.0,  155.0, 250.0,  156.0, 430.0,  165.0, 400.0,  160.0, 240.0,  120.0, 240.0  );  stipe.setFill(stipeColor);  stipe.setStroke(Color.BLACK);  stipe.setStrokeType(StrokeType.OUTSIDE);  return stipe;  }  static public Polygon getUpperPartOfStipe() {  Polygon stipe = new Polygon(  160.0, 247.0,  170.0, 245.0,  173.0, 240.0,  160.0, 210.0,  157.0, 185.0,  120.0, 185.0,  110.0, 225.0,  100.0, 237.0,  100.0, 240.0,  120.0, 247.0,  130.0, 243.0,  140.0, 255.0,  150.0, 245.0,  155.0, 250.0  );  stipe.setFill(stipeColor);  stipe.setStroke(Color.BLACK);  stipe.setStrokeType(StrokeType.OUTSIDE);  return stipe;  }  static public Polygon getCap() {  Polygon stipe = new Polygon(  157.0, 188.0,  197.0, 183.0,  220.0, 175.0,  240.0, 155.0,  250.0, 140.0,  250.0, 135.0,  240.0, 125.0,  220.0, 115.0,  220.0, 115.0,  195.0, 100.0,  180.0, 92.0,  150.0, 92.0,  128.0, 102.0,  90.0, 128.0,  50.0, 150.0,  45.0, 153.0,  42.0, 163.0,  42.0, 175.0,  120.0, 185.0  );  stipe.setFill(capColorPrimary);  stipe.setStroke(Color.BLACK);  stipe.setStrokeType(StrokeType.OUTSIDE);  return stipe;  }  static public Ellipse[] getCapStains() {  Ellipse[] stains = new Ellipse[10];  stains[0] = getStain(150.0, 110.0, 15, 5);  stains[1] = getStain(130.0, 130.0, 14, 6);  stains[2] = getStain(110.0, 150.0, 13, 5);  stains[3] = getStain(170.0, 135.0, 12, 6);  stains[4] = getStain(205.0, 120.0, 11, 6);  stains[5] = getStain(150.0, 160.0, 12, 7);  stains[6] = getStain(115.0, 170.0, 13, 6);  stains[7] = getStain(190.0, 165.0, 14, 5);  stains[8] = getStain(210.0, 145.0, 15, 6);  stains[9] = getStain(70.0, 165.0, 14, 7);  return stains;  }  static private Ellipse getStain(double x, double y, double w, double h) {  Ellipse stain = new Ellipse(x, y, w, h);  stain.setFill(stipeColor);  stain.setStroke(Color.BLACK);  stain.setStrokeType(StrokeType.OUTSIDE);  return stain;  }  } |

**Результат**

|  |  |
| --- | --- |
|  |  |
|  |  |