

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”

|  |  |  |  |
| --- | --- | --- | --- |
| Виконала  студентка III курсу  групи КП-81  Мозгова Катерина Олегівна  (*прізвище, ім’я, по батькові*)  варіант № 12 |  |  | Зарахована  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладачем  Шкурат Оксаною Сергіївною (*прізвище, ім’я, по батькові*) |

Київ 2021

**Завдання за варіантом**

**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:**



# 

**Код програми**

|  |
| --- |
| **Main.java** |
| package sample;  import javafx.application.Application; import javafx.stage.Stage; import javafx.scene.Scene;  import javafx.scene.Group; //import java.awt.Color;  import javafx.animation.\*;  import javafx.scene.paint.Color; import javafx.scene.shape.Circle; import javafx.scene.shape.Ellipse; import javafx.scene.shape.Polygon; import javafx.util.Duration;  public class Main extends Application{  Color appBack = Color.*rgb*(255,128,64);  Color bodyAndHead = Color.*rgb*(66, 135, 245);  Color bellyAndEyes = Color.*rgb*(255, 255, 255);  Color wings = Color.*rgb*(7, 0, 224);  Color feetAndBeak = Color.*rgb*(255, 221, 0);  Color pupils = Color.*rgb*(0, 0, 0);   int width = 700;  int height = 700;  int cx = width / 2;  int cy = height / 2;   public static void main(String[] args) {  *launch*(args);  }   @Override  public void start(Stage primaryStage) {  Group root = new Group();  Scene scene = new Scene(root, width, height);  scene.setFill(appBack);   //draw penguin  drawPenguin(root);   //animation  int time = 3000;   //rotate  RotateTransition rotate = new RotateTransition(Duration.*millis*(time), root);  rotate.setByAngle(360f);  rotate.setCycleCount(Timeline.*INDEFINITE*);   //scale  ScaleTransition scaleFrom = new ScaleTransition(Duration.*millis*(time), root);  scaleFrom.setToX(1);  scaleFrom.setToY(1);   ScaleTransition scaleTo = new ScaleTransition(Duration.*millis*(time), root);  scaleTo.setToX(0);  scaleTo.setToY(0);   SequentialTransition scale = new SequentialTransition();  scale.getChildren().addAll(  scaleTo,  scaleFrom  );  scale.setCycleCount(Timeline.*INDEFINITE*);   //translate  TranslateTransition translateTo = new TranslateTransition(Duration.*millis*(time), root);  translateTo.setFromY(0);  translateTo.setToY(250);  translateTo.setCycleCount(Timeline.*INDEFINITE*);  translateTo.setAutoReverse(true);   TranslateTransition translateFrom = new TranslateTransition(Duration.*millis*(time), root);  translateFrom.setFromY(250);  translateFrom.setToY(0);  translateFrom.setCycleCount(Timeline.*INDEFINITE*);  translateFrom.setAutoReverse(true);   SequentialTransition translate = new SequentialTransition();  translate.getChildren().addAll(  translateTo,  translateFrom  );  translate.setCycleCount(Timeline.*INDEFINITE*);   ParallelTransition animation = new ParallelTransition();  animation.getChildren().addAll(  rotate,  scale,  translate  );  animation.play();  //   primaryStage.setTitle("Penguin");  primaryStage.setScene(scene);  primaryStage.show();  }   public void drawPenguin(Group root) {  Circle head = new Circle(cx, cy - 70, 50, bodyAndHead);  Ellipse body = new Ellipse(cx, cy, 60, 70);  body.setFill(bodyAndHead);   Ellipse belly = new Ellipse(cx, cy, 35, 50);  belly.setFill(bellyAndEyes);   Ellipse leftWing = new Ellipse(cx - 50, cy - 10, 10, 30);  leftWing.setFill(wings);  Ellipse rightWing = new Ellipse(cx + 50, cy - 10, 10, 30);  rightWing.setFill(wings);   Ellipse leftFoot = new Ellipse(cx - 25, cy + 65, 20, 15);  leftFoot.setFill(feetAndBeak);  Ellipse rightFoot = new Ellipse(cx + 25, cy + 65, 20, 15);  rightFoot.setFill(feetAndBeak);   Polygon beak = new Polygon(  cx - 20, cy - 60,  cx + 20, cy - 60,  cx, cy - 35  );  beak.setFill(feetAndBeak);   Ellipse leftEye = new Ellipse(cx - 10, cy - 70, 13, 20);  leftEye.setFill(bellyAndEyes);  Ellipse rightEye = new Ellipse(cx + 18, cy - 65, 18, 13);  rightEye.setFill(bellyAndEyes);   Circle leftPupil = new Circle(cx - 5, cy - 65, 5, pupils);  Circle rightPupil = new Circle(cx + 8, cy - 65, 5, pupils);   root.getChildren().add(head);  root.getChildren().add(body);  root.getChildren().add(belly);  root.getChildren().add(leftWing);  root.getChildren().add(rightWing);  root.getChildren().add(leftFoot);  root.getChildren().add(rightFoot);  root.getChildren().add(beak);  root.getChildren().add(leftEye);  root.getChildren().add(rightEye);  root.getChildren().add(leftPupil);  root.getChildren().add(rightPupil);  } } |

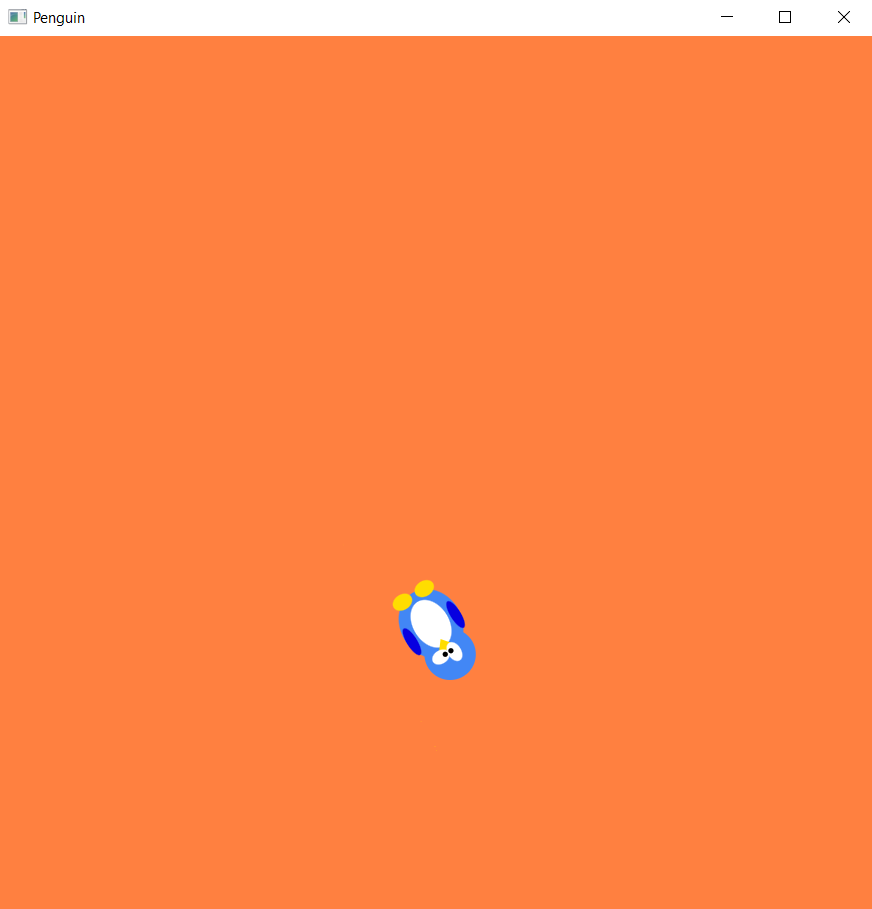
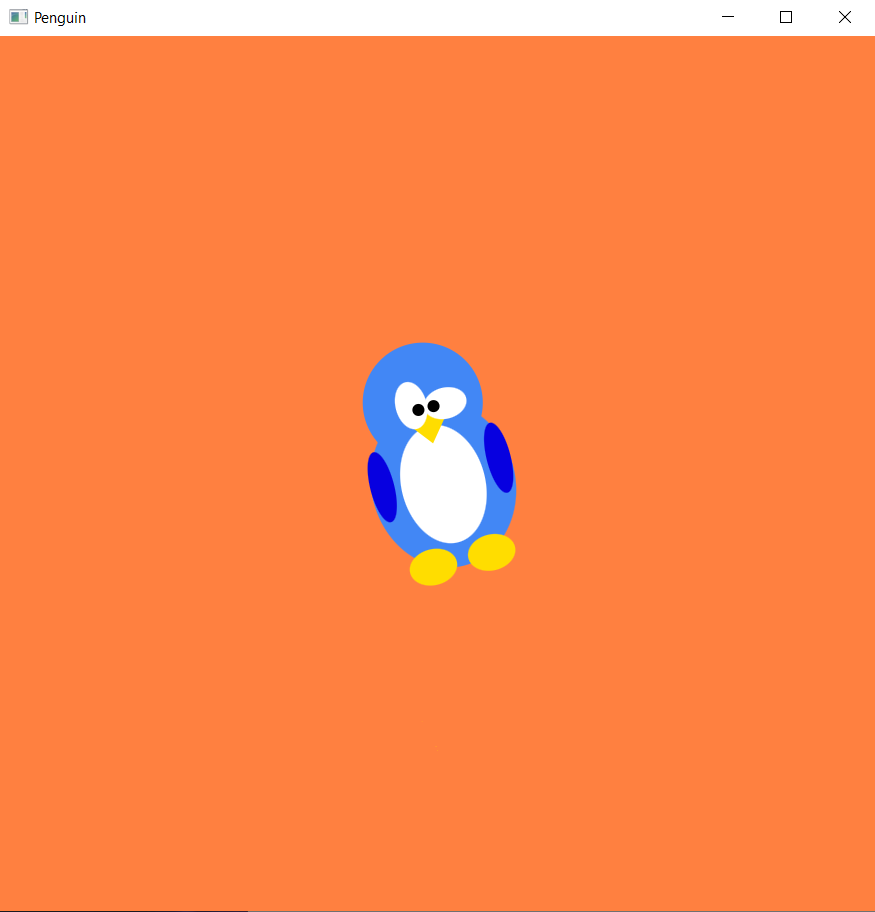
**Результат**

Рис. 2. Результат роботи програми