|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Отчет по выполнению лабораторной работы 5

**Тема. СОЗДАНИЕ ПРОГРАМ С ГРАФИЧЕСКИМ ИНТЕРФЕЙСОМ ПОЛЬЗОВАТЕЛЯ НА JAVA**

Дисциплина Программирование на языке Джава

Выполнил

|  |  |
| --- | --- |
| студент | Болотов Михаил |
|  | Фамилия И.О. |
| группа | ИКБО-06-19 |
|  | Номер группы |

Москва 2020

**Содержание**

[​ Теория 2](#__RefHeading___Toc4128_811265792)

[​ Важное замечание 2](#__RefHeading___Toc28006_811265792)

[​ Задание 2](#__RefHeading___Toc4136_811265792)

[​ Код 3](#__RefHeading___Toc4138_811265792)

[​ Circle.java 3](#__RefHeading___Toc28008_811265792)

[​ ShapeTest.java 3](#__RefHeading___Toc28010_811265792)

[​ Square.java 5](#__RefHeading___Toc28012_811265792)

[​ Shape.java 6](#__RefHeading___Toc28014_811265792)

[​ Triangle.java 7](#__RefHeading___Toc28016_811265792)

[​ Скриншот 8](#__RefHeading___Toc3748_811265792)

[​ Заключение 8](#__RefHeading___Toc3750_811265792)

[​ Библиографический список 9](#__RefHeading___Toc3752_811265792)

# Теория

Для создания графического интерфейса пользователя можно использовать стандартную Java библиотеку Swing или AWT. В этих библиотеках имеются различные классы, позволяющие создавать окна, кнопки, текстовые поля, меню и другие объекты.

Text Fields - текстовое поле или поля для ввода текста (можно ввести только одну строку). Примерами текстовых полей являются поля для ввода логина и пароля, например, используемые, при входе в электронную почту.

Пример содания объекта класса JTextField:

JTextField jta = new JTextField (10);

В параметре конструктора задано число 10, это количество символов, которые могут быть видны в текстовом поле. Текст веденный в поле JText может быть возвращен с помощью метода getText(). Также в поле можно записать новое значение с помощью метода setText(String s).

Как и у других компонентов, мы можем изменять цвет и шрифт текста в текстовом поле.

### Важное замечание

Ответственность за выполнение проверки на наличие ошибок в коде лежит полностью на программисте, например, чтобы проверить произойдет ли ошибка, когда в качестве входных данных в JTextField ожидается ввод числа. Компилятор не будет ловить такого рода ошибку, поэтому ее необходимо обрабатывать пользовательским кодом.

# Задание

Создать окно, нарисовать в нем 20 случайных фигур, случайного цвета. Классы фигур должны наследоваться от абстрактного класса Shape, в котором описаны свойства фигуры: цвет, позиция.

# Код

### Circle.java

package dev.ky3he4ik.lab.lab5;  
  
import java.awt.\*;  
import java.awt.geom.Ellipse2D;  
  
public class Circle extends Shape {  
 protected int radius;  
  
 public Circle(int radius, int x, int y, Color color) {  
 super(x, y, color);  
 this.radius = radius;  
 }  
  
 public Circle() {  
 radius = 0;  
 }  
  
 @Override  
 public void draw(Graphics g) {  
 Graphics2D g2d = (Graphics2D) g;  
 Ellipse2D.Double circle = new Ellipse2D.Double(x, y, radius \* 2, radius \* 2);  
  
 g2d.setColor(color);  
 g2d.fill(circle);  
 }  
}

### ShapeTest.java

package dev.ky3he4ik.lab.lab5;  
  
import javax.swing.\*;  
import java.awt.\*;  
import java.util.ArrayList;  
import java.util.Random;  
  
public class ShapeTest extends JPanel {  
 private static final int N\_FIGURES = 40;  
 private ArrayList<Shape> shapes;  
  
 ShapeTest() {  
 shapes = new ArrayList<>(N\_FIGURES);  
 Random random = new Random();  
 for (int i = 0; i < N\_FIGURES; i++) {  
 int side = random.nextInt(100) + 20;  
 int x = random.nextInt(640);  
 int y = random.nextInt(480);  
 Color color = new Color(random.nextInt(), false);  
 switch (random.nextInt(3)) {  
 case 0:  
 shapes.add(new Circle(side, x, y, color));  
 break;  
 case 1:  
 shapes.add(new Triangle(side, x, y, color));  
 break;  
 case 2:  
 shapes.add(new Square(side, x, y, color));  
 break;  
 }  
 }  
 setBackground(Color.BLACK);  
 }  
  
 @Override  
 public Dimension getPreferredSize() {  
 return new Dimension(640, 480);  
 }  
  
 @Override  
 public void paintComponent(Graphics g) {  
 super.paintComponents(g);  
 for (Shape shape : shapes)  
 shape.draw(g);  
  
 }  
  
 public static void main(String[] args) {  
 SwingUtilities.invokeLater(new Runnable() {  
 @Override  
 public void run() {  
 ShapeTest mainPanel = new ShapeTest();  
 JFrame frame = new JFrame();  
// frame.setSize(640, 480);  
 frame.setBackground(Color.BLACK);  
 frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  
 frame.getContentPane().add(mainPanel);  
 frame.pack();  
 frame.setLocationByPlatform(true);  
 frame.setVisible(true);  
 }  
 });  
 new ShapeTest();  
 }  
}

### Square.java

package dev.ky3he4ik.lab.lab5;  
  
import java.awt.\*;  
import java.awt.geom.GeneralPath;  
  
public class Square extends Shape {  
 protected int side;  
  
 public Square(int side, int x, int y, Color color) {  
 super(x, y, color);  
 this.side = side;  
 }  
  
 public Square() {  
 }  
  
 public double getSide() {  
 return side;  
 }  
  
 public void setSide(int side) {  
 this.side = side;  
 }  
  
 @Override  
 public void draw(Graphics g) {  
 int[] xPoints = {x, side + x, side + x, x};  
 int[] yPoints = {y, y, side + y, side + y};  
  
 g.setColor(color);  
 g.fillPolygon(xPoints, yPoints, xPoints.length);  
 }  
}

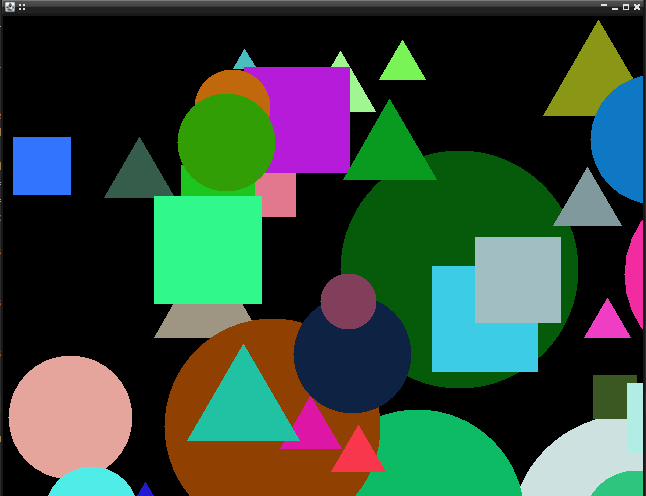
### Shape.java

package dev.ky3he4ik.lab.lab5;  
  
import java.awt.\*;  
import java.awt.geom.GeneralPath;  
  
abstract public class Shape {  
 protected int x;  
 protected int y;  
 protected Color color;  
  
 public Shape(int x, int y, Color color) {  
 this.x = x;  
 this.y = y;  
 this.color = color;  
 }  
  
 public Shape() {  
 x = 0;  
 y = 0;  
 color = Color.BLACK;  
 }  
  
 public int getX() {  
 return x;  
 }  
  
 public void setX(int x) {  
 this.x = x;  
 }  
  
 public int getY() {  
 return y;  
 }  
  
 public void setY(int y) {  
 this.y = y;  
 }  
  
 public Color getColor() {  
 return color;  
 }  
  
 public void setColor(Color color) {  
 this.color = color;  
 }  
  
 public abstract void draw(Graphics g);  
  
 protected void drawPath(Graphics g, double[] xPoints, double[] yPoints) {  
 Graphics2D g2d = (Graphics2D) g;  
 GeneralPath figure = new GeneralPath();  
  
 figure.moveTo(xPoints[0] + x, yPoints[0] + y);  
 for (int i = 1; i < xPoints.length; i++)  
 figure.lineTo(xPoints[i] + x, yPoints[i] + y);  
  
 figure.closePath();  
  
 g2d.setColor(Color.YELLOW);  
 g2d.fill(figure);  
 }  
}

### Triangle.java

package dev.ky3he4ik.lab.lab5;  
  
import java.awt.\*;  
  
public class Triangle extends Shape {  
 protected int side;  
  
 public Triangle(int side, int x, int y, Color color) {  
 super(x, y, color);  
 this.side = side;  
 }  
  
 public Triangle() {  
 }  
  
 public double getSide() {  
 return side;  
 }  
  
 public void setSide(int side) {  
 this.side = side;  
 }  
  
 @Override  
 public void draw(Graphics g) {  
 int[] xPoints = {x, side + x, (side / 2 + x)};  
 int[] yPoints = {(int) (side \* 0.866 + y), (int) (side \* 0.866 + y), y}; // sqrt(3)/2 = 0.8660...  
  
 g.setColor(color);  
 g.fillPolygon(xPoints, yPoints, xPoints.length);  
 }  
}

# Скриншот



# Заключение

В данной лабораторной работе я научился создавать графический интерфейс пользователя, освоил на практике работу с различными объектами для создания ГИП, менеджерами размещения компонентов.

# Библиографический список

1. Зорина Н.В. Курс лекций по Объектно-ориентированному программированию на Java, МИРЭА, Москва, 2016
2. Программирование на языке Java: работа со строками и массивами. Методические указания. [Электронный ресурс] : Учебно-методические пособия — Электрон. дан. — СПб. : ПГУПС, 2015. — 24 с.
3. Кожомбердиева, Г.И. Программирование на языке Java: создание графического интерфейса пользователя: учеб. пособие. [Электронный ресурс] : Учебные пособия / Г.И. Кожомбердиева, М.И. Гарина. — Электрон. дан. — СПб.: ПГУПС, 2012. — 67 с.
4. Вишневская, Т.И. Технология программирования. Часть 1. [Электронный ресурс] / Т.И. Вишневская, Т.Н. Романова. — Электрон. дан. — М. : МГТУ им. Н.Э. Баумана, 2007. — 59 с.