**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Факультет Информационных технологий**

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 9**

**Дисциплина:** Технологии кроссплатформенного программирования

**Тема: Создание меню, графика в языке Java.**

**Выполнила: студентка группы 211-725**

Алюбаева Карина Ислямбековна

**Дата**  03.12.2023

**Москва**

**2023**

Оглавление

[Введение 2](#_Toc152521685)

[Цель работы: 2](#_Toc152521686)

[Задание: 2](#_Toc152521687)

[Теоретическое обоснование 3](#_Toc152521688)

[Ход работы 4](#_Toc152521689)

[Выполнение Задания 1. 4](#_Toc152521690)

[Результат работы: 10](#_Toc152521691)

[Выполнение Задания 2 16](#_Toc152521692)

[Листинг 16](#_Toc152521693)

[Результат работы программы: 28](#_Toc152521694)

# Введение

## Цель работы:

Научиться работать с меню и графикой на Java.

## Задание:

1. Переделать пример с рисованием фигур так, чтобы фигуры рисовались выбором соответствующего пункта меню.
2. Нарисовать свою фамилию линиями разной толщины, цвета и стиля.

# Теоретическое обоснование

Для создания меню в языке Java есть удобный класс JMenuBar, который представляет собой строку меню, куда можно вставлять пункты меню с нужными подпунктами. Для пунктов меню и подпунктов можно использовать класс JMenuItem, вставляя подпункты в соответствующие пункты меню. Чтобы установить созданное меню на нужное окно, нужно использовать метод setJMenuBar для выбранного объекта типа JFrame. Для использования графических возможностей языка Java необходимо создать свой компонент, на котором будет происходить рисование, унаследовав его от стандартного компонента пакета Swing, и переопределить в нем метод paintComponent.

# Ход работы

## Выполнение Задания 1.

//Файл JMyPanel.java

import java.awt.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.JPanel;

public class JMyPanel extends JPanel {

public static enum Figure {

Линия, Круг, Квадрат, Закругленный\_квадрат, Очистить

};

private Figure vibor = Figure.Очистить;

public JMyPanel() {

}

public void ris(String s) {

vibor = Figure.valueOf(s);

repaint();

}

public void paintComponent(Graphics gr) {

super.paintComponent(gr);

Graphics2D g = (Graphics2D) gr;

BasicStroke pen;

g.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,

RenderingHints.VALUE\_ANTIALIAS\_ON);

switch (vibor) {

case Линия:

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setColor(Color.blue);

g.drawLine(20, 20, 100, 100);

break;

case Круг:

float[] dash = { 10, 30 };

pen = new BasicStroke(10, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_ROUND, 10, dash, 0);

g.setStroke(pen);

g.setColor(Color.red);

g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.green, true));

g.fill(new Ellipse2D.Double(20, 20, 100, 100));

break;

case Квадрат:

float[] dash2 = { 20, 20 };

pen = new BasicStroke(5, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_BEVEL, 1, dash2, 0);

g.setStroke(pen);

g.setColor(Color.magenta);

g.drawRect(20, 20, 100, 100);

break;

case Закругленный\_квадрат:

float[] dash3 = { 20, 20, 2, 20, 2, 20 };

pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_BEVEL, 1, dash3, 0);

g.setStroke(pen);

g.setColor(Color.yellow);

g.drawRoundRect(20, 20, 100, 100, 60, 60);

break;

case Очистить:

g.clearRect(0, 0, getSize().width, getSize().height);

break;

}

}

}

import java.awt.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.JPanel;

public class JMyPanel extends JPanel {

    public static enum Figure {

        Линия, Круг, Квадрат, Закругленный\_квадрат, Очистить

    };

    private Figure vibor = Figure.Очистить;

    public JMyPanel() {

    }

    public void ris(String s) {

        vibor = Figure.valueOf(s);

        repaint();

    }

    public void paintComponent(Graphics gr) {

        super.paintComponent(gr);

        Graphics2D g = (Graphics2D) gr;

        BasicStroke pen;

        g.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,

                RenderingHints.VALUE\_ANTIALIAS\_ON);

        switch (vibor) {

            case Линия:

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setColor(Color.blue);

                g.drawLine(20, 20, 100, 100);

            break;

            case Круг:

                float[] dash = { 10, 30 };

                pen = new BasicStroke(10, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_ROUND, 10, dash, 0);

                g.setStroke(pen);

                g.setColor(Color.red);

                g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.green, true));

                g.fill(new Ellipse2D.Double(20, 20, 100, 100));

            break;

            case Квадрат:

                float[] dash2 = { 20, 20 };

                pen = new BasicStroke(5, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_BEVEL, 1, dash2, 0);

                g.setStroke(pen);

                g.setColor(Color.magenta);

                g.drawRect(20, 20, 100, 100);

            break;

            case Закругленный\_квадрат:

                float[] dash3 = { 20, 20, 2, 20, 2, 20 };

                pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_BEVEL, 1, dash3, 0);

                g.setStroke(pen);

                g.setColor(Color.yellow);

                g.drawRoundRect(20, 20, 100, 100, 60, 60);

            break;

            case Очистить:

                g.clearRect(0, 0, getSize().width, getSize().height);

            break;

        }

    }

}

//Файл MyGraph.java

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class MyGraph extends JFrame implements ActionListener {

private JMyPanel myPanel = new JMyPanel();

public static void main(String[] args) {

new MyGraph("Лабораторная работа №9");

}

public MyGraph(String s) {

super(s);

JMenuBar menu = new JMenuBar();

JMenu first = new JMenu("Фигуры");

menu.add(first);

add(menu, BorderLayout.NORTH);

JMenuItem[] figs = new JMenuItem[5];

for (int i = 0; i < 5; i++) {

figs[i] = new JMenuItem(JMyPanel.Figure.values()[i].toString());

figs[i].addActionListener(this);

first.add(figs[i]);

}

Dimension dim = new Dimension(500, 500);

setMinimumSize(dim);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

add(myPanel, BorderLayout.CENTER);

pack();

setVisible(true);

}

public void actionPerformed(ActionEvent e) {

myPanel.ris(e.getActionCommand());

}

}

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class MyGraph extends JFrame implements ActionListener {

    private JMyPanel myPanel = new JMyPanel();

    public static void main(String[] args) {

        new MyGraph("Лабораторная работа №9");

    }

    public MyGraph(String s) {

        super(s);

        JMenuBar menu = new JMenuBar();

        JMenu first = new JMenu("Фигуры");

        menu.add(first);

        add(menu, BorderLayout.NORTH);

        JMenuItem[] figs = new JMenuItem[5];

        for (int i = 0; i < 5; i++) {

            figs[i] = new JMenuItem(JMyPanel.Figure.values()[i].toString());

            figs[i].addActionListener(this);

            first.add(figs[i]);

        }

        Dimension dim = new Dimension(500, 500);

        setMinimumSize(dim);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        add(myPanel, BorderLayout.CENTER);

        pack();

        setVisible(true);

    }

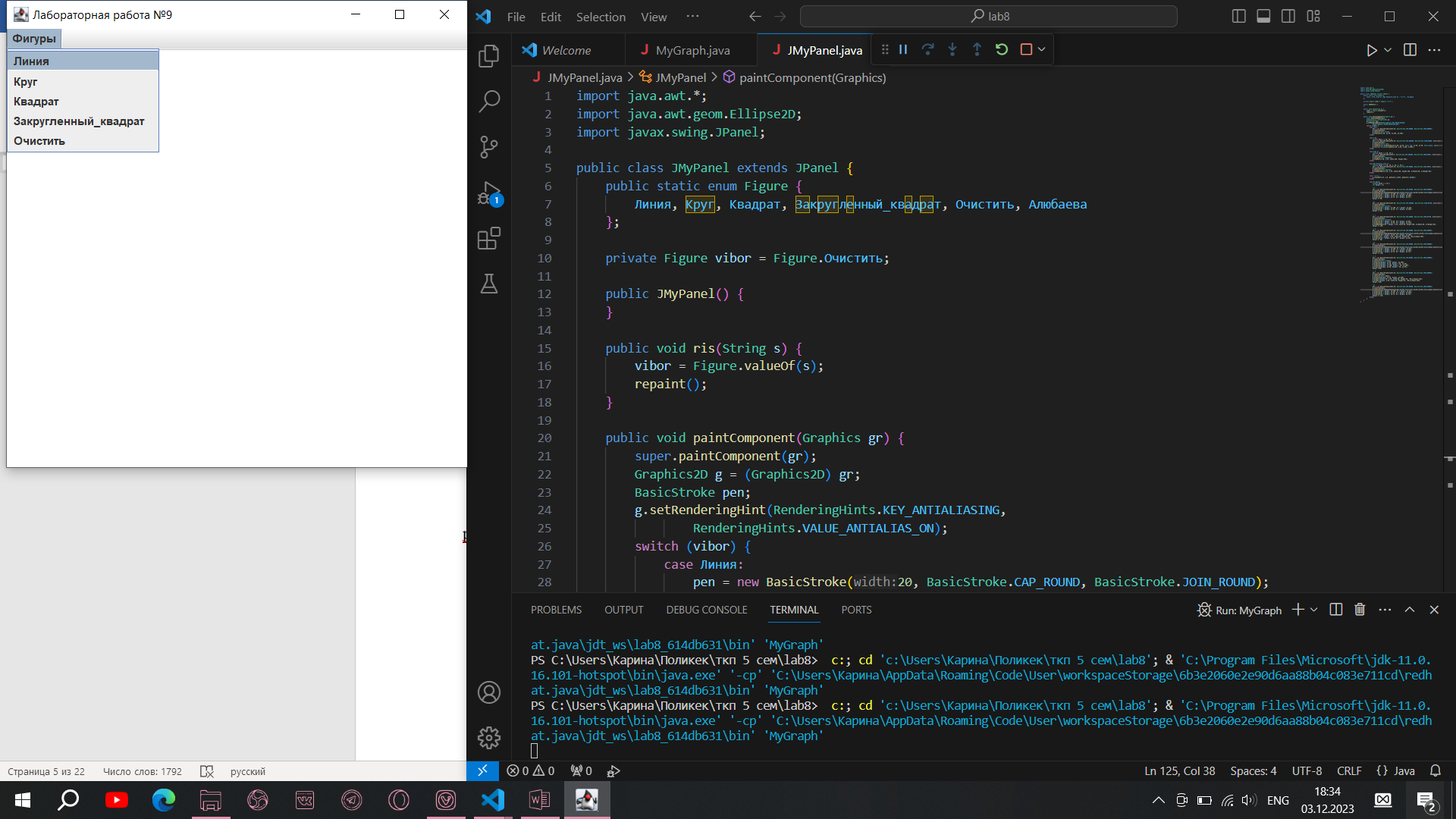
    public void actionPerformed(ActionEvent e) {

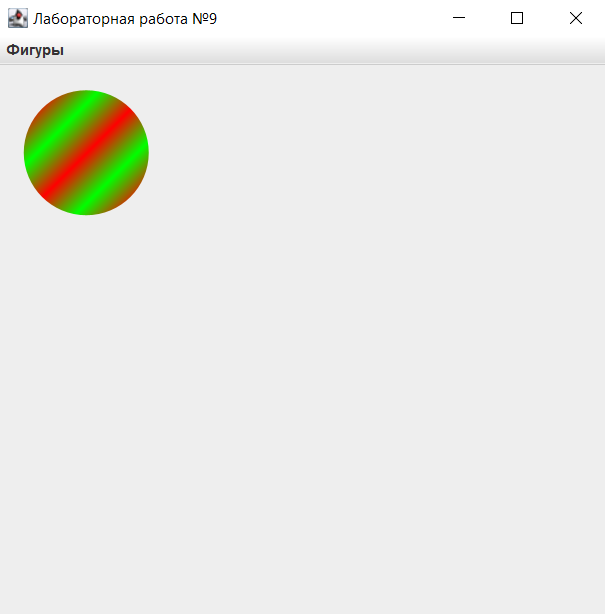
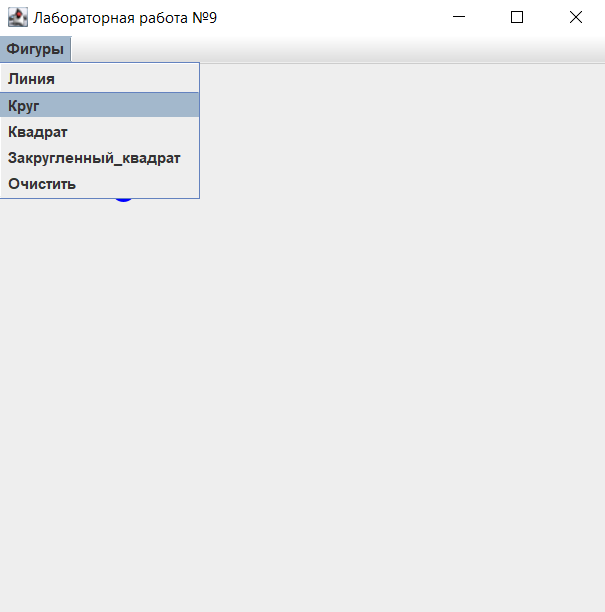
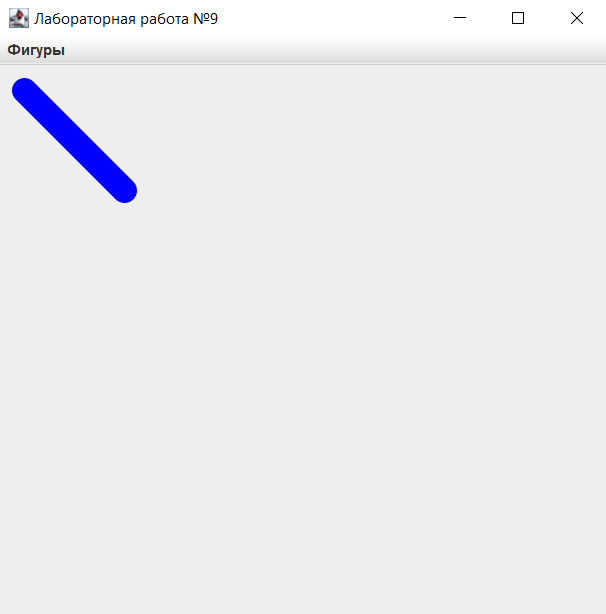
        myPanel.ris(e.getActionCommand());

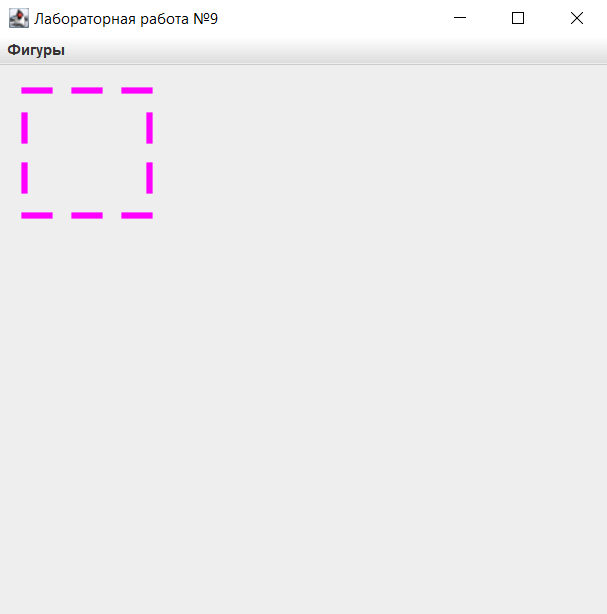
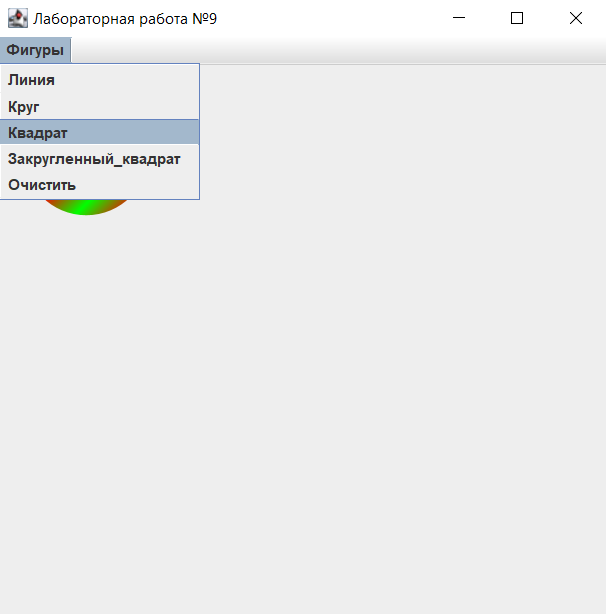
    }

}

### Результат работы:







## Выполнение Задания 2

### Листинг

Я изменила файлы из первого задания, добавив еще один пункт меню, куда при нажатии выводится моя фамилия, написанная интересным способом.

Изменения в коде, вызванные добавлением слушателей, я выделила маркером, чтобы лучше было заметно. Остальное осталось таким, каким было раньше.

//Файл JMyPanel.java

import java.awt.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.JPanel;

public class JMyPanel extends JPanel {

public static enum Figure {

Линия, Круг, Квадрат, Закругленный\_квадрат, Очистить, Алюбаева

};

private Figure vibor = Figure.Очистить;

public JMyPanel() {

}

public void ris(String s) {

vibor = Figure.valueOf(s);

repaint();

}

public void paintComponent(Graphics gr) {

super.paintComponent(gr);

Graphics2D g = (Graphics2D) gr;

BasicStroke pen;

g.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,

RenderingHints.VALUE\_ANTIALIAS\_ON);

switch (vibor) {

case Линия:

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setColor(Color.blue);

g.drawLine(20, 20, 100, 100);

break;

case Круг:

float[] dash = { 10, 30 };

pen = new BasicStroke(10, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_ROUND, 10, dash, 0);

g.setStroke(pen);

g.setColor(Color.red);

g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.green, true));

g.fill(new Ellipse2D.Double(20, 20, 100, 100));

break;

case Квадрат:

float[] dash2 = { 20, 20 };

pen = new BasicStroke(5, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_BEVEL, 1, dash2, 0);

g.setStroke(pen);

g.setColor(Color.magenta);

g.drawRect(20, 20, 100, 100);

break;

case Закругленный\_квадрат:

float[] dash3 = { 20, 20, 2, 20, 2, 20 };

pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_BEVEL, 1, dash3, 0);

g.setStroke(pen);

g.setColor(Color.yellow);

g.drawRoundRect(20, 20, 100, 100, 60, 60);

break;

case Очистить:

g.clearRect(0, 0, getSize().width, getSize().height);

break;

case Алюбаева:

float[] dash4 = { 10 };

int margin = 0;

//А

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

g.drawLine(20 + margin, 100, 47 + margin, 20);

g.drawLine(47 + margin, 20, 75 + margin, 100);

g.drawLine(28 + margin, 75, 65 + margin, 75);

margin += 90;

//Л

pen = new BasicStroke(20, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_MITER, 1, dash4, 0);

g.setStroke(pen);

g.setColor(Color.green);

g.drawLine(20 + margin, 100, 47 + margin, 20);

g.drawLine(47 + margin, 20, 75 + margin, 100);

margin += 90;

// Ю

pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_MITER, 1, dash4, 0);

g.setStroke(pen);

g.setColor(Color.yellow);

g.drawLine(50 + margin, 60, 20 + margin, 60);

g.drawLine(20 + margin, 20, 20 + margin, 110);

g.drawRoundRect(50 + margin, 20, 50, 100, 60, 60);

margin += 105;

// Б

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setPaint(new GradientPaint(30, 30, Color.blue, 50, 50, Color.pink, true));

g.drawLine(20 + margin, 20, 20 + margin, 100);

g.drawArc(10 + margin, 50, 50, 50, -90,180);

g.drawLine(20 + margin, 20, 60 + margin, 20);

margin += 80;

//А

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

g.drawLine(20 + margin, 100, 47 + margin, 20);

g.drawLine(47 + margin, 20, 75 + margin, 100);

g.drawLine(28 + margin, 75, 65 + margin, 75);

margin += 110;

// Е

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setColor(Color.orange);

g.drawLine(margin, 20, margin, 100);

g.drawLine(margin, 20, margin + 50, 20);

g.drawLine(margin, 100, margin + 50, 100);

g.drawLine(margin, 50, margin + 40, 50);

margin +=90;

// В

pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setColor(Color.pink);

g.drawLine(margin, 20, margin, 100);

g.drawArc(margin-10, 50, 50, 50, -90,180);

g.drawArc(margin, 20, 30, 30, -90,180);

margin +=50;

//А

pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

g.setStroke(pen);

g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

g.drawLine(20 + margin, 100, 47 + margin, 20);

g.drawLine(47 + margin, 20, 75 + margin, 100);

g.drawLine(28 + margin, 75, 65 + margin, 75);

margin += 110;

break;

}

}

}

import java.awt.\*;

import java.awt.geom.Ellipse2D;

import javax.swing.JPanel;

public class JMyPanel extends JPanel {

    public static enum Figure {

        Линия, Круг, Квадрат, Закругленный\_квадрат, Очистить, Алюбаева

    };

    private Figure vibor = Figure.Очистить;

    public JMyPanel() {

    }

    public void ris(String s) {

        vibor = Figure.valueOf(s);

        repaint();

    }

    public void paintComponent(Graphics gr) {

        super.paintComponent(gr);

        Graphics2D g = (Graphics2D) gr;

        BasicStroke pen;

        g.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,

                RenderingHints.VALUE\_ANTIALIAS\_ON);

        switch (vibor) {

            case Линия:

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setColor(Color.blue);

                g.drawLine(20, 20, 100, 100);

            break;

            case Круг:

                float[] dash = { 10, 30 };

                pen = new BasicStroke(10, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_ROUND, 10, dash, 0);

                g.setStroke(pen);

                g.setColor(Color.red);

                g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.green, true));

                g.fill(new Ellipse2D.Double(20, 20, 100, 100));

            break;

            case Квадрат:

                float[] dash2 = { 20, 20 };

                pen = new BasicStroke(5, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_BEVEL, 1, dash2, 0);

                g.setStroke(pen);

                g.setColor(Color.magenta);

                g.drawRect(20, 20, 100, 100);

            break;

            case Закругленный\_квадрат:

                float[] dash3 = { 20, 20, 2, 20, 2, 20 };

                pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_BEVEL, 1, dash3, 0);

                g.setStroke(pen);

                g.setColor(Color.yellow);

                g.drawRoundRect(20, 20, 100, 100, 60, 60);

            break;

            case Очистить:

                g.clearRect(0, 0, getSize().width, getSize().height);

            break;

            case Алюбаева:

                float[] dash4 = { 10 };

                int margin = 0;

                //А

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

                g.drawLine(20 + margin, 100, 47 + margin, 20);

                g.drawLine(47 + margin, 20, 75 + margin, 100);

                g.drawLine(28 + margin, 75, 65 + margin, 75);

                margin += 90;

                //Л

                pen = new BasicStroke(20, BasicStroke.CAP\_SQUARE, BasicStroke.JOIN\_MITER, 1, dash4, 0);

                g.setStroke(pen);

                g.setColor(Color.green);

                g.drawLine(20 + margin, 100, 47 + margin, 20);

                g.drawLine(47 + margin, 20, 75 + margin, 100);

                margin += 90;

                // Ю

                pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_MITER, 1, dash4, 0);

                g.setStroke(pen);

                g.setColor(Color.yellow);

                g.drawLine(50 + margin, 60, 20 + margin, 60);

                g.drawLine(20 + margin, 20, 20 + margin, 110);

                g.drawRoundRect(50 + margin, 20, 50, 100, 60, 60);

                margin += 105;

                // Б

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setPaint(new GradientPaint(30, 30, Color.blue, 50, 50, Color.pink, true));

                g.drawLine(20 + margin, 20, 20 + margin, 100);

                g.drawArc(10 + margin, 50, 50, 50, -90,180);

                g.drawLine(20 + margin, 20, 60 + margin, 20);

                margin += 80;

                //А

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

                g.drawLine(20 + margin, 100, 47 + margin, 20);

                g.drawLine(47 + margin, 20, 75 + margin, 100);

                g.drawLine(28 + margin, 75, 65 + margin, 75);

                margin += 110;

                // Е

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setColor(Color.orange);

                g.drawLine(margin, 20, margin, 100);

                g.drawLine(margin, 20, margin + 50, 20);

                g.drawLine(margin, 100, margin + 50, 100);

                g.drawLine(margin, 50, margin + 40, 50);

                margin +=90;

                // В

                pen = new BasicStroke(10, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setColor(Color.pink);

                g.drawLine(margin, 20, margin, 100);

                g.drawArc(margin-10, 50, 50, 50, -90,180);

                g.drawArc(margin, 20, 30, 30, -90,180);

                margin +=50;

                //А

                pen = new BasicStroke(20, BasicStroke.CAP\_ROUND, BasicStroke.JOIN\_ROUND);

                g.setStroke(pen);

                g.setPaint(new GradientPaint(30, 30, Color.red, 50, 50, Color.pink, true));

                g.drawLine(20 + margin, 100, 47 + margin, 20);

                g.drawLine(47 + margin, 20, 75 + margin, 100);

                g.drawLine(28 + margin, 75, 65 + margin, 75);

                margin += 110;

            break;

        }

    }

}

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class MyGraph extends JFrame implements ActionListener {

private JMyPanel myPanel = new JMyPanel();

public static void main(String[] args) {

new MyGraph("Лабораторная работа №9");

}

public MyGraph(String s) {

super(s);

JMenuBar menu = new JMenuBar();

JMenu first = new JMenu("Фигуры");

JMenu second = new JMenu("Фамилия");

JMenuItem imya = new JMenuItem("Алюбаева");

menu.add(first);

menu.add(second);

second.add(imya);

imya.addActionListener(this);

add(menu, BorderLayout.NORTH);

JMenuItem[] figs = new JMenuItem[5];

for (int i = 0; i < 5; i++) {

figs[i] = new JMenuItem(JMyPanel.Figure.values()[i].toString());

figs[i].addActionListener(this);

first.add(figs[i]);

}

Dimension dim = new Dimension(500, 500);

setMinimumSize(dim);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

add(myPanel, BorderLayout.CENTER);

pack();

setVisible(true);

}

public void actionPerformed(ActionEvent e) {

myPanel.ris(e.getActionCommand());

}

}

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class MyGraph extends JFrame implements ActionListener {

    private JMyPanel myPanel = new JMyPanel();

    public static void main(String[] args) {

        new MyGraph("Лабораторная работа №9");

    }

    public MyGraph(String s) {

        super(s);

        JMenuBar menu = new JMenuBar();

        JMenu first = new JMenu("Фигуры");

        JMenu second = new JMenu("Фамилия");

        JMenuItem imya = new JMenuItem("Алюбаева");

        menu.add(first);

        menu.add(second);

        second.add(imya);

        imya.addActionListener(this);

        add(menu, BorderLayout.NORTH);

        JMenuItem[] figs = new JMenuItem[5];

        for (int i = 0; i < 5; i++) {

            figs[i] = new JMenuItem(JMyPanel.Figure.values()[i].toString());

            figs[i].addActionListener(this);

            first.add(figs[i]);

        }

        Dimension dim = new Dimension(500, 500);

        setMinimumSize(dim);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        add(myPanel, BorderLayout.CENTER);

        pack();

        setVisible(true);

    }

    public void actionPerformed(ActionEvent e) {

        myPanel.ris(e.getActionCommand());

    }

}

### Результат работы программы:

