****Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Кафедра автоматизації проектування енергетичних процесів і систем

ЗВІТ   
про виконання лабораторної роботи №6  
з дисципліни «Технології конструювання програмного забезпечення»

**«**Построение графических интерфейсов с помощью инструментов Swing **»**

Виконав: студент групи TI-92

Черноусов Денис Ігорович

Перевірив: доцент, к.ф.-м.н.  
Тарнавський Ю.А.

Київ - 2020

# Ход выполнения работы

1. Создайте проект **Java**.
2. Добавьте в проект форму и разместите на ней компоненты в соответствии с вариантом индивидуального задания.
3. Отредактируйте файл класса формы, обеспечив функциональность программы в соответствии с вариантом индивидуального задания.
4. Добавьте unit-тесты для тестирования методов класса, воспользовавшись командой **Code-Generate…-Test...**, и выполните их тестирование.
5. Если обнаружены методы, не прошедшие тест, внесите необходимые изменения в код и повторите тест.
6. Вставьте необходимые комментарии и сгенерируйте справку Javadoc.

**16 варіант**

5. Вывод строки заданным шрифтом и заданного цвета в графическом окне. В верхней панели (JPanel) "Управление выводом" задается надпись (JLabel) "Текст:" и текстовое поле (JTextField), надпись (JLabel) "Гарнитура:", вращающийся список (JSpinner) со значениями "Times New Roman" (шрифт по умолчанию), "Arial" и "Verdana", надпись (JLabel) "Цвет:" и вращающийся список (JSpinner) со значениями "Черный" (цвет по умолчанию), "Красный", "Зеленый" и "Синий", а также кнопка (JButton) "Вывести строку". В нижней панели (JPanel) "Вывод строки" выводится в графическом контексте (с помощью метода drawString()) в произвольном месте пустая строка. При задании текста строки в текстовом поле, параметров строки во вращающихся списках верхней панели и нажатии кнопки "Вывод строки" строка заданного содержимого и цвета выводится заданным шрифтом в нижней панели.

13. Ввод текста в графическое окно приложения. В верхней панели (JPanel) окна "Характеристики шрифта" определены следующие компоненты: надпись (JLabel) "Цвет:", раскрывающееся меню (JComboBox), в котором четыре пункта: "Черный" (цвет по умолчанию), "Красный", "Зеленый" и "Синий", надпись (JLabel) "Размер:" и три радиокнопки (JRadioButton) с надписями: "12pt" (размер по умолчанию), "14pt" и "16pt". В нижней текстовой панели (JTextPane) "Ввод текста" окна приложения вводится текст, набираемый на клавиатуре. При выборе одного из пунктов меню текст в панели выводится соответствующим цветом и/или соответствующего размера.

20. Изменение фигуры в графическом окне. В центре верхней панели (JPanel) "Вывод треугольника" в графическом контексте рисуется (с помощью метода drawPolygon()) треугольник. В нижней панели (JPanel) "Параметры треугольника" задаются следующие компоненты: надпись (JLabel) "Цвет:", вращающийся список (JSpinner) со значениями: "Черный" (цвет по умолчанию), "Красный", "Зеленый" и "Синий", надпись (JLabel) "Номер вершины:", вращающийся список (JSpinner) со значениями "1", "2" и "3", надпись (JLabel) "Координата X:", текстовое поле (JTextField), надпись (JLabel) "Координата Y:", текстовое поле (JTextField) и кнопка (JButton) "Изменить треугольник". При наборе данных во вращающихся списках (в пикселях), вводе данных в текстовых полях (в пикселях) и при нажатии кнопки "Изменить треугольник" треугольник перерисовывается заданным цветом и с заданным новым положением одной из вершин. Перед выводом фигуры выполняется проверка, введены ли в текстовых полях все данные и являются ли они целыми числами.

25. Ввод текста в графическое окно приложения. В верхней панели окна (JPanel) "Параметры шрифта" заданы следующие компоненты: надпись (JLabel) "Цвет:", четыре радиокнопки (JRadioButton) с надписями: "Черный" (цвет по умолчанию), "Красный", "Зеленый" и "Синий", надпись (JLabel) "Размер:" и текстовое поле (JTextField) (начальное значение этого текстового поля равно 12). В текстовой панели (JTextPane) "Ввод текста" окна приложения вводится текст, набираемый на клавиатуре. Надписи для радиокнопок должны быть выполнены соответствующим цветом. При выборе одной из радиокнопок цвета и ввода в текстовом поле целого числа – размера текста содержимое текстовой панели выводится соответствующим цветом и/или соответствующего размера. При вводе значения в текстовое поле проверяется, является ли это значение целым числом, содержащим одну или две цифры.

Допоміжний клас Colorist

package Tasks;  
  
import java.awt.\*;  
  
public class Colorist {  
 public static Color getColor(Object MyColor) {  
  
 return (MyColor.toString() == "Black") ? Color.*BLACK* :  
 (MyColor.toString() == "Green") ? Color.*GREEN* :  
 (MyColor.toString() == "Red") ? Color.*RED* :  
 (MyColor.toString() == "Blue") ? Color.*BLUE* : Color.*BLACK*;  
 }  
  
}

Завдання 5

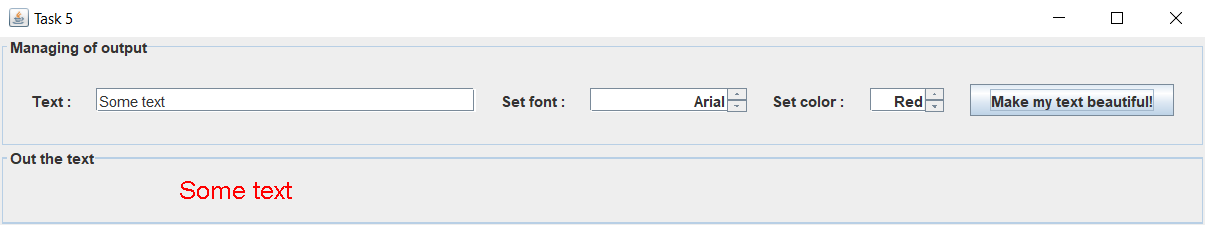
Клас MyFrame

package Tasks.task5;  
  
  
import Tasks.Colorist;  
  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import javax.swing.\*;  
import javax.swing.border.Border;  
  
  
public class MyFrame extends JFrame implements ActionListener {  
  
 private JPanel PanelHigh = new JPanel(new FlowLayout(FlowLayout.*LEFT*, 20, 20));  
 private JPanel PanelLow = new JPanel(new FlowLayout(FlowLayout.*LEFT*, 20, 20));  
 private JTextField TextField;  
 private JSpinner MyFont;  
 private JSpinner MyColor;  
 private JButton Button;  
  
  
 MyFrame() {  
  
 super("Task 5");  
 setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*);  
 setVisible(true);  
  
  
 PanelHigh.setName("Managing of output");  
 Border border = BorderFactory.*createTitledBorder*(PanelHigh.getName());  
 PanelHigh.setBorder(border);  
  
  
 PanelHigh.add(new JLabel("Text :"));  
 TextField = new JTextField("Some text", 30);  
 PanelHigh.add(TextField);  
  
  
 PanelHigh.add(new JLabel("Set font :"));  
 MyFont = new JSpinner(new SpinnerListModel(new String[]{"Times New Roman", "Arial", "Verdana"}));  
 PanelHigh.add(MyFont);  
  
 PanelHigh.add(new JLabel("Set color :"));  
 MyColor = new JSpinner(new SpinnerListModel(new String[]{"Black", "Red", "Green", "Blue"}));  
 MyColor.setPreferredSize(new Dimension(60 ,20));  
 PanelHigh.add(MyColor);  
  
 Button = new JButton("Make my text beautiful!");  
 PanelHigh.add(Button);  
  
 add(PanelHigh, BorderLayout.*PAGE\_START*);  
  
  
 PanelLow.setName("Out the text");  
 border = BorderFactory.*createTitledBorder*(PanelLow.getName());  
 PanelLow.setBorder(border);  
  
 add(PanelLow);  
  
 Button.addActionListener(this);  
  
  
 pack();  
 }  
  
  
 public void actionPerformed(ActionEvent event) {  
  
 Graphics g = getGraphics();  
  
 System.*out*.println(getTypeOfString());  
  
 paint(g);  
  
 }  
  
 public String getTypeOfString() {  
 return MyFont.getValue().toString() + " - " + MyColor.getValue().toString();  
 }  
  
 public void paint(Graphics g) {  
 super.paint(g);  
  
 g.setColor(Colorist.*getColor*(MyColor.getValue()));  
  
 g.setFont(new Font(MyFont.getValue().toString(), Font.*PLAIN*, 20));  
  
 g.drawString(TextField.getText(), 150, 160 );  
 }  
  
}

Клас Main

package Tasks.task5;  
  
  
public class Main {  
 public static void main(String[] args) {  
 new MyFrame();  
 }  
}

Вивід



Завдання 13

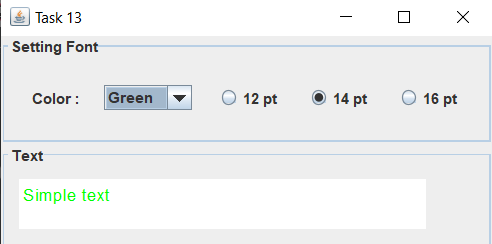
Клас GUI

package Tasks.task13;  
  
import Tasks.Colorist;  
import jdk.jfr.Event;  
  
import javax.swing.\*;  
import javax.swing.border.Border;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
  
  
public class GUI extends JFrame implements ActionListener {  
 private JPanel Panel1 = new JPanel(new FlowLayout(FlowLayout.*LEFT*, 20, 20));  
 private ButtonGroup buttonGroup = new ButtonGroup();  
 private JComboBox ComboBox;  
 private JRadioButton Button12;  
 private JRadioButton Button14;  
 private JRadioButton Button16;  
 private JPanel Panel2 = new JPanel(new FlowLayout(FlowLayout.*LEFT*, 10, 10));  
 private JTextPane TextPane;  
  
 public GUI() {  
  
 super("Task 13");  
 setVisible(true);  
 setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*);  
  
  
 Panel1.setName("Setting Font");  
 Border border = BorderFactory.*createTitledBorder*(Panel1.getName());  
 Panel1.setBorder(border);  
  
 Panel1.add(new JLabel("Color :"));  
 ComboBox = new JComboBox(new String[]{"Black", "Red", "Green", "Blue"});  
 ComboBox.setPreferredSize(new Dimension(70, 20));  
 Panel1.add(ComboBox);  
  
 Panel1.add(Button12 = new JRadioButton("12 pt"));  
 Button12.setActionCommand("12");  
 Panel1.add(Button14 = new JRadioButton("14 pt"));  
 Button14.setActionCommand("14");  
 Panel1.add(Button16 = new JRadioButton("16 pt"));  
 Button16.setActionCommand("16");  
 buttonGroup.add(Button12);  
 buttonGroup.add(Button14);  
 buttonGroup.add(Button16);  
 Button12.setSelected(true);  
  
  
 add(Panel1, BorderLayout.PAGE\_START);  
  
  
 Panel2.setName("Text");  
 border = BorderFactory.*createTitledBorder*(Panel2.getName());  
 Panel2.setBorder(border);  
  
 TextPane = new JTextPane();  
 TextPane.setText("Simple text");  
 TextPane.setFont(new Font("Arial", Font.*PLAIN*, 12));  
 TextPane.setPreferredSize(new Dimension(325, 40));  
 Panel2.add(TextPane);  
  
 add(Panel2);  
  
 Button12.addActionListener(this);  
 Button14.addActionListener(this);  
 Button16.addActionListener(this);  
 ComboBox.addActionListener(this);  
  
  
 pack();  
 }  
  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
 System.*out*.println(getSettings());  
  
 TextPane.setForeground(Colorist.*getColor*(ComboBox.getSelectedItem()) );  
  
 TextPane.setFont(new Font("Arial", Font.*PLAIN*, Integer.*parseInt*(  
 buttonGroup.getSelection().getActionCommand())));  
  
 }  
  
  
 public String getSettings(){  
 return ComboBox.getSelectedItem() + " - " + buttonGroup.getSelection().getActionCommand() + "pt";  
 }  
  
}

Клас Main

package task13;  
  
public class Main {  
 public static void main(String[] args) {  
  
 Measurable o1 = new Quadrant();  
  
 System.*out*.println(o1);  
  
 o1.setSide(-1);  
  
 System.*out*.println(o1);  
  
 o1.setSide(10);  
  
 System.*out*.println(o1);  
  
 Measurable o2 = (Quadrant) o1.clone();  
  
 System.*out*.println(o2);  
  
  
 }  
}

Вивід



Завдання 20

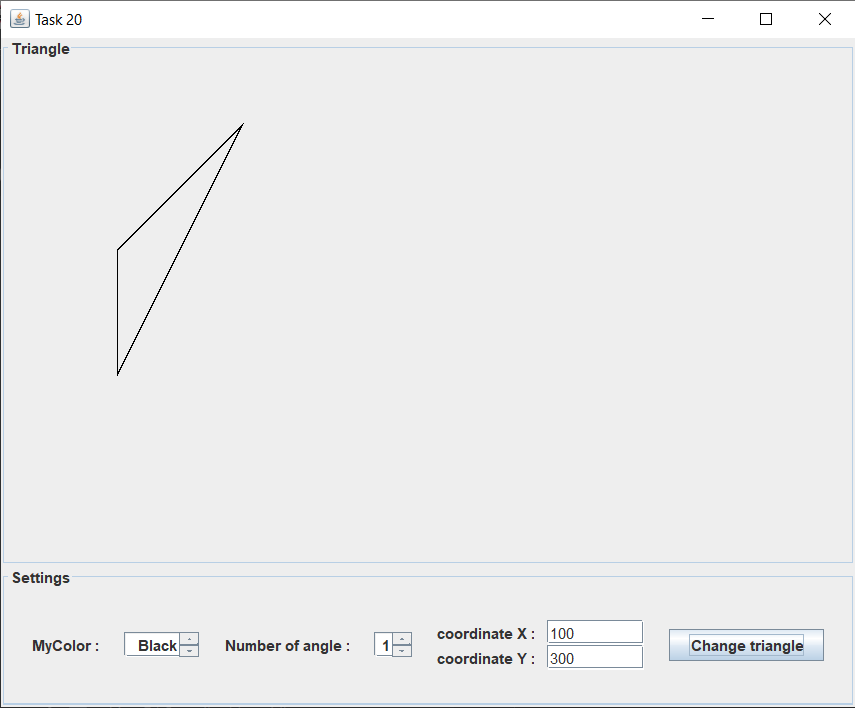
Клас GUI

package Tasks.task20;  
  
import Tasks.Colorist;  
  
import javax.swing.\*;  
import javax.swing.border.Border;  
import java.awt.\*;  
import java.awt.event.\*;  
import java.util.Arrays;  
  
public class GUI extends JFrame implements ActionListener {  
 private JPanel Triangle = new JPanel(new FlowLayout(FlowLayout.*CENTER*, 0, 200));  
 private JPanel Parameters = new JPanel(new FlowLayout(FlowLayout.*CENTER*, 20, 25));  
 private JPanel Grid = new JPanel(new GridLayout(2, 2, 10, 0));  
 private JSpinner MyColor;  
 private JSpinner MyAngle;  
 private JTextField FieldX;  
 private JTextField FieldY;  
 private JButton Button;  
  
 private int arrayX[] = new int[]{100, 100, 200};  
 private int arrayY[] = new int[]{100, 200, 100};  
  
  
 GUI() {  
 super("Task 20");  
 setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*);  
 setVisible(true);  
  
  
 Triangle.setName("Triangle");  
 Border border = BorderFactory.*createTitledBorder*(Triangle.getName());  
 Triangle.setBorder(border);  
  
 add(Triangle, BorderLayout.*PAGE\_START*);  
  
  
 Parameters.setName("Settings");  
 border = BorderFactory.*createTitledBorder*(Parameters.getName());  
 Parameters.setBorder(border);  
  
 Parameters.add(new JLabel("MyColor :"));  
 MyColor = new JSpinner(new SpinnerListModel(new String[]{"Black", "Red", "Green", "Blue"}));  
 MyColor.setPreferredSize(new Dimension(60, 20));  
 Parameters.add(MyColor);  
  
 Parameters.add(new JLabel("Number of angle :"));  
 MyAngle = new JSpinner(new SpinnerNumberModel(1, 1, 3, 1));  
 Parameters.add(MyAngle);  
  
 Grid.add(new JLabel("coordinate X :"));  
  
 Grid.add(FieldX);  
  
 Grid.add(new JLabel("coordinate Y :"));  
  
 Grid.add(FieldY);  
  
 Parameters.add(Grid);  
  
 Button = new JButton("Change triangle");  
 Parameters.add(Button);  
  
  
 add(Parameters);  
  
  
 Button.addActionListener(this);  
  
 pack();  
  
 }  
  
 @Override  
 public void actionPerformed(ActionEvent e) {  
  
  
 if (!FieldX.getText().matches("[0-9]+") || !FieldY.getText().matches("[0-9]+")) {  
 JOptionPane.*showMessageDialog*(null, "Error! Enter integer numbers!");  
 }  
  
 arrayX[(int) MyAngle.getValue() - 1] = Integer.*parseInt*(FieldX.getText());  
 arrayY[(int) MyAngle.getValue() - 1] = Integer.*parseInt*(FieldY.getText());  
  
 paint(getGraphics());  
  
 System.*out*.println(this);  
  
 }  
  
 public void paint(Graphics g) {  
 super.paint(g);  
  
 g.setColor(Colorist.*getColor*(MyColor));  
  
 g.drawPolygon(arrayX, arrayY, 3);  
 }  
  
 @Override  
 public String toString() {  
 return "arrayX=" + Arrays.*toString*(arrayX) +  
 "\n" +  
 "arrayY=" + Arrays.*toString*(arrayY)+  
 "\n";  
 }  
}

Клас Main

package Tasks.task20;  
  
public class Main {  
 public static void main(String[] args) {  
 new GUI();  
 }  
}

Вивід



Завдання 25

Клас GUI

package Tasks.task25;  
  
import javax.swing.\*;  
import javax.swing.border.Border;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
public class GUI extends JFrame implements ActionListener {  
 private JRadioButton radioBlack = new JRadioButton("Black");  
 private JRadioButton radioRed = new JRadioButton("Red");  
 private JRadioButton radioGreen = new JRadioButton("Green");  
 private JRadioButton radioBlue = new JRadioButton("Blue");  
 private JTextField SizeField = new JTextField();  
 private JTextPane TextPane = new JTextPane();  
 private JPanel LowPanel = new JPanel( new FlowLayout( FlowLayout.*CENTER* , 40 ,40));  
 private JPanel Grid = new JPanel(new GridLayout(2,2));  
 private JPanel HighPanel = new JPanel( new FlowLayout( FlowLayout.*CENTER* , 20 ,20));  
 private ButtonGroup Group = new ButtonGroup();  
  
  
 public GUI() {  
 super("Task 25");  
 setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*);  
 setVisible(true);  
  
 HighPanel.setName("Font settings");  
 Border border = BorderFactory.*createTitledBorder*(HighPanel.getName());  
 HighPanel.setBorder(border);  
  
  
 HighPanel.add(new JLabel("Color :"));  
  
 radioBlack.setSelected(true);  
 radioBlack.setForeground(Color.*BLACK*);  
 Group.add(radioBlack);  
 Grid.add(radioBlack);  
  
 radioRed.setForeground(Color.*RED*);  
 Group.add(radioRed);  
 Grid.add(radioRed);  
  
 radioGreen.setForeground(Color.*GREEN*);  
 Group.add(radioGreen);  
 Grid.add(radioGreen);  
  
 radioBlue.setForeground(Color.*BLUE*);  
 Group.add(radioBlue);  
 Grid.add(radioBlue);  
  
 HighPanel.add(Grid);  
  
 HighPanel.add (new JLabel("Size :"));  
 SizeField.setText("12");  
 SizeField.setPreferredSize(new Dimension(20, 20));  
 HighPanel.add(SizeField);  
  
  
 add (HighPanel , BorderLayout.*PAGE\_START*) ;  
  
  
 LowPanel.setName("Out the text");  
 border = BorderFactory.*createTitledBorder*(LowPanel.getName());  
 LowPanel.setBorder(border);  
  
 TextPane.setText("Simple text");  
 LowPanel.add(TextPane);  
  
 add(LowPanel);  
  
  
 radioBlack.addActionListener(this);  
 radioBlue.addActionListener(this);  
 radioRed.addActionListener(this);  
 radioGreen.addActionListener(this);  
  
 SizeField.addActionListener(this) ;  
  
 pack();  
  
 }  
  
 public void actionPerformed(ActionEvent e) {  
  
 TextPane.setForeground((radioGreen.isSelected())? Color.*GREEN* :  
 (radioRed.isSelected())? Color.*RED* :  
 (radioBlue.isSelected())? Color.*BLUE* :  
 (radioBlack.isSelected())? Color.*BLACK* : Color.*BLACK*);  
  
  
 if (!SizeField.getText().matches("[1-9][0-9]") ) {  
 JOptionPane.*showMessageDialog*(null, "Error! Enter integer number between 10 and 99!");  
 SizeField.setText("12");  
 }  
  
 TextPane.setFont( new Font (null , Font.*PLAIN* , Integer.*parseInt*(SizeField.getText())));  
 }  
  
  
  
  
}

Клас Main

package Tasks.task25;  
  
public class Main {  
 public static void main(String[] args) {  
 new GUI();  
 }  
}

Вивід

