Національний Технічний Університет України

«Київський Політехнічний Інститут»

Лабораторна робота №5

Дизайн та реалізація інтерфейсу користувача

Виконав

Студент 2-го курсу ФІОТ

групи ІО-92

Зубко Павло Сергійович

Дата здачі: **06.04.2011**

Київ 2011

**Завдання**

**1. Вивчити принципи побудови графічного інтерфейсу. Вільно володіти поняттями компонента, контейнера та графічного контексту.**

**2. Ознайомитись з бібліотеками графічних інтерфейсів AWT та SWING. Вільно володіти поняттями “важкого” та “легкого” компонентів.**

**3. Ознайомитись з моделлю малювання в Java. Вільно володіти призначенням методів paint(),update() та repaint().**

**4. Розробити клас DiagramDrawer для відображення табличних даних у вигляді діаграми (згідно варіанту). Клас має містити метод draw(Graphics g) який здійснює малювання в графічному контексті g.**

**5. Створити тестовий клас TestDraw. В методі main():**

**завантажити табличні дані з файлу і перевірити їх коректність (ЛР 2-3);**

**створити екземпляр класу DiagramDrawer і ініціювати його завантаженними даними;**

**створити екземпляр класу Frame, в якому екземпляр DiagramDrawer буде використовуватись в якості стратегії для малювання.**

**Лістинг**

**package** com.lab111.labwork2;

**import** java.awt.\*;

**import** javax.swing.JPanel;

**public** **class** DiagramDrawer **implements** Drawer{

/\*\*

\*

\*/

**private** **static** **final** **long** *serialVersionUID* = 1L;

**private** Point P;

**private** CSVProcessor csv;

**private** JPanel panel;

DiagramDrawer(CSVProcessor csv,**int** x,**int** y, **int** width, **int** height)

{

panel=**new** JPanel();

panel.setLocation(x, y);

panel.setSize(width, height);

P=**new** Point(x,y);

**this**.csv=csv;

}

**public** **void** Draw(Graphics g)

{

**this**.panel.paint(g);

**double** sum=0;

String[][] matrix=csv.getMatrix();

**for**(**int** i=0;i<matrix.length;i++)

sum+=Double.*parseDouble*(matrix[i][1]);

**int** red=255;

**int** green=0;

**int** blue=0;

**for**(**int** i=0;i<matrix.length;i++){

Color c=**new** Color(red,green,blue);

g.setColor(c);

**int** deltax = 10;

**int** deltay = (**int**) Math.*round*(Integer.*parseInt*(matrix[i][1])/sum);

g.drawLine((**int**)P.getX(),(**int**)P.getY(),(**int**)P.getX()+deltax, (**int**)P.getX()+deltay);

P.setLocation(P.getX()+deltax, P.getY()+deltay);

red=(red+50)%255;

green=(green+20)%255;

blue=(blue+10)%255;

}

}

**public** JPanel GetPanel()

{

**return** **this**.panel;

}

}

package com.lab111.labwork2;

import javax.swing.\*;

import java.awt.\*;

public class MyFrame extends JFrame {

private DiagramDrawer dd;

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

MyFrame(DiagramDrawer dd) {

super();

int w = Toolkit.getDefaultToolkit().getScreenSize().width;

int h = Toolkit.getDefaultToolkit().getScreenSize().height;

this.setSize(w / 2, h / 2);

this.setTitle("Курсовой проект");

this.setLocation(w / 4, h / 4);

this.setResizable(true);

this.setDefaultCloseOperation(EXIT\_ON\_CLOSE);

this.setVisible(true);

this.dd=dd;

this.add(dd.GetPanel());

}

public void paint(Graphics g)

{

super.paint(g);

this.dd.Draw(g);

}

}

package com.lab111.labwork2;

import java.awt.Button;

import java.awt.Graphics;

import javax.swing.JButton;

public class MyButton extends Button{

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

MyButton(){

super();

}

public void paint(Graphics g)

{

super.paint(g);

this.setLocation(30,30);

this.setSize(150, 130);

this.setVisible(true);

}

}

package com.lab111.labwork2;

import java.awt.Dimension;

import java.awt.Graphics;

import javax.swing.JTable;

public class MyTable extends JTable{

/\*\*

\*

\*/

private static final long serialVersionUID = 1572517175816549160L;

public MyTable(String[][] matrix,String[] columns)

{

super(matrix,columns);

}

public void paintComponent (Graphics g)

{

super.paintComponent(g);

//this.setLocation(this.getParent().WIDTH/2,this.getParent().HEIGHT/2);

this.setLocation(20,20);

this.setMaximumSize(new Dimension(320,240));

//this.setBounds(50, 50, 200, 200);

for (int i=0;i<this.getColumnCount();i++)

{

this.getColumnModel().getColumn(i).setMaxWidth(90);

}

//this.setVisible(true);

}

}