**РУСЕНСКИ УНИВЕРСИТЕТ “АНГЕЛ КЪНЧЕВ”**

Катедра: **Компютърни системи и технологии**

КУРСОВА РАБОТА  
ПО ПРОГРАМНИ ЕЗИЦИ

Студент: **Бейтула Хамди Реджеб**

Факултетен номер: **103217**

Група: **26Б**

Специалност: **КСТ**

Дата: Изготвил:  
 Проверил:

/ гл.ас. инж. Ю. Ст. Златева /

# Задание.

Да се разработи на програмния език Java елементарен векторен графичен редактор.

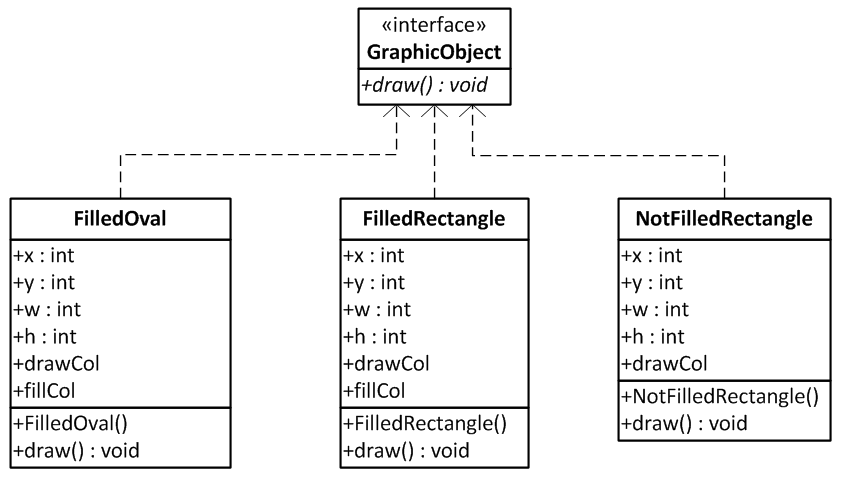
* Графичният редактор да позволява избор на цвят за изчертаване и евентуално на цвят за запълване и да поддържа следните графични примитиви: ***незапълнен четириъгълник, запълнен четириъгълник и запълнен овал;***
* Изборът на примитив, който ще се въвежда, да става ***списък (java.awt.List);***
* Потребителският интерфейс и самите графични примитиви да са разположени в ***един и същ прозорец;***
* По време на работа с редактора обектите от графичните примитиви да се съхраняват в ***HashMap***. В потребителския интерфейс да се предвиди команда за извеждане на съдържанието му в конзола;
* Да се използва пакетът ***awt;***
* Как се задават примитивите е по избор на студента;
* В програмата може да се използва различни класове по усмотрение на студента, но задължително за всеки графичен примитив трябва да има отделен клас. Всички тези класове трябва да реализират следния интерфейс:

*public interface GraphicObject {*

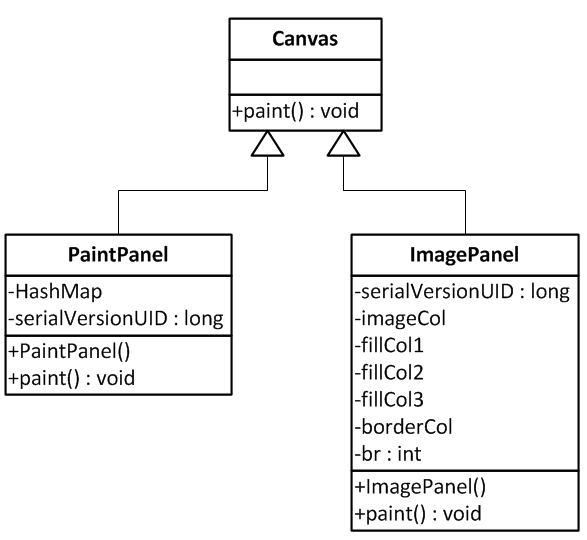
*public abstract void draw();*

*}*

# Клас диаграми.



Фиг. 1 Имплементиране на GraphicObject.



Фиг. 2 Наследяване на Canvas.

# Листинг.

* **Main.java**

**package mypack;**

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

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

**import java.util.HashMap;**

**import java.util.Map;**

**public class Main {**

**static int hashMapKey, intNumPr;**

**Color curDCol, curFCol, curBGCol;**

**HashMap<String, GraphicObject> hm;**

**int curPr, msgB;**

**Main(int s) {**

**msgB = s;**

**}**

**/\* Конструктор \*/**

**Main() {**

**final List prList = new List();**

**Button btnDraw = new Button("Изчертай");**

**Button btnHashMapToCons = new Button("Изведи");**

**final Label numPr = new Label("0");**

**Panel ctrlPanel = new Panel();**

**Panel drawPanel = new Panel();**

**Label lPr = new Label("Примитиви:");**

**final Label lNum = new Label("Брой елементи:");**

**final TextField x1 = new TextField(4);**

**final TextField x2 = new TextField(4);**

**final TextField y1 = new TextField(4);**

**final TextField y2 = new TextField(4);**

**hm = new HashMap<String, GraphicObject>();**

**Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();**

**Frame f = new Frame("Графичен редактор - 103217");**

**f.setLayout(new FlowLayout(FlowLayout.LEFT, 0, 0));**

**f.setBounds((screenSize.width - 1000) / 2,**

**((screenSize.height - 600) / 2) - 100, 1000, 600);**

**f.setResizable(false);**

**f.setBackground(new Color(200, 200, 200));**

**f.addWindowListener(new WindowAdapter () {**

**public void windowClosing(WindowEvent e) {**

**System.exit(0);**

**}**

**});**

**intNumPr = curPr = 0;**

**hashMapKey = Integer.MAX\_VALUE;**

**curBGCol = Color.white;**

**curDCol = curFCol = Color.black;**

**ctrlPanel.setLayout(null);**

**ctrlPanel.setSize(200, f.getHeight());**

**ctrlPanel.setBackground(new Color(190, 190, 190));**

**final int maxCtrlPanW = ctrlPanel.getWidth();**

**lPr.setBounds(5, 0, maxCtrlPanW, 20);**

**ctrlPanel.add(lPr);**

**prList.add("Незапълнен четириъгълник");**

**prList.add("Запълнен четириъгълник");**

**prList.add("Запълнен овал");**

**prList.setBounds(0, 20, maxCtrlPanW, 50);**

**prList.addItemListener(new ItemListener() {**

**public void itemStateChanged(ItemEvent ee) {**

**String clickedItem = (String) prList.getSelectedItem();**

**if (clickedItem.equals("Незапълнен четириъгълник")) curPr = 0;**

**else if (clickedItem.equals("Запълнен четириъгълник")) curPr = 1;**

**else if (clickedItem.equals("Запълнен овал")) curPr = 2;**

**}**

**});**

**ctrlPanel.add(prList);**

**final PaintPanel paintPanel = new PaintPanel(hm);**

**/\* Избор на цвят \*/**

**final ImagePanel cDr = new ImagePanel();**

**cDr.setBounds(0, 75, maxCtrlPanW, 66);**

**ctrlPanel.add(cDr);**

**cDr.addMouseListener(new MouseAdapter(){**

**public void mouseClicked(MouseEvent m) {**

**int y = m.getY(), x = m.getX();**

**if (y >= 25 && y <= 65) {**

**int clr = cDr.imageCol.getRGB(x, y-25);**

**int r = (clr & 0x00ff0000) >> 16;**

**int g = (clr & 0x0000ff00) >> 8;**

**int b = clr & 0x000000ff;**

**if (ImagePanel.br == 0) {**

**curDCol = new Color(r, g, b);**

**cDr.fillCol1 = curDCol;**

**ImagePanel.br++;**

**}**

**else if (ImagePanel.br == 1){**

**curFCol = new Color(r, g, b);**

**cDr.fillCol2 = curFCol;**

**ImagePanel.br++;**

**}**

**else if (ImagePanel.br == 2){**

**curBGCol = new Color(r, g, b);**

**cDr.fillCol3 = curBGCol;**

**paintPanel.setBackground(curBGCol);**

**ImagePanel.br = 0;**

**}**

**cDr.repaint();**

**}**

**}**

**});**

**/\* Бутон - Изчертай \*/**

**btnDraw.setBounds(90, 290, 105, 25);**

**btnDraw.addActionListener(new ActionListener(){**

**public void actionPerformed(ActionEvent e) {**

**int a, b, c, d;**

**try {**

**a = Integer.parseInt(x1.getText());**

**b = Integer.parseInt(y1.getText());**

**c = Integer.parseInt(x2.getText());**

**d = Integer.parseInt(y2.getText());**

**if (a > 783 || a < 0)**

**messageFrame("Моля въведете цяло число между 0 и 783 за Х1");**

**else if (b > 560 || b < 0)**

**messageFrame("Моля въведете цяло число между 0 и 560 за Y1");**

**else if (c > 783 || (c + a) > 783 || c < 0)**

**messageFrame("Моля въведете цяло и коректно число за Х2");**

**else if (d > 560 || (d + b) > 560 || d < 0)**

**messageFrame("Моля въведете цяло и коректно число за Y2");**

**else {**

**if (curPr == 0)**

**hm.put(Integer.toString(hashMapKey), new NotFilledRectangle(a, b, c, d, curDCol));**

**else if (curPr == 1)**

**hm.put(Integer.toString(hashMapKey), new FilledRectangle(a, b, c, d, curDCol, curFCol));**

**else if (curPr == 2)**

**hm.put(Integer.toString(hashMapKey), new FilledOval(a, b, c, d, curDCol, curFCol));**

**paintPanel.repaint();**

**hashMapKey--;**

**intNumPr++;**

**numPr.setText(Integer.toString(intNumPr));**

**x1.setText(null); y1.setText(null);**

**x2.setText(null); y2.setText(null);**

**ImagePanel.br = 0;**

**cDr.repaint();**

**}**

**}**

**catch(Exception exc){**

**messageFrame("Грешна или непопълнена стойност!");**

**}**

**}**

**});**

**ctrlPanel.add(btnDraw);**

**/\* Бутон - Извеждане \*/**

**btnHashMapToCons.setBounds(5, 290, 80, 25);**

**btnHashMapToCons.addActionListener(new ActionListener(){**

**public void actionPerformed(ActionEvent e) {**

**if (hm.size() > 0)**

**for (Map.Entry<String, GraphicObject> me : hm.entrySet())**

**System.out.println("Ключ: " + me.getKey() + ", Стойност: " + me.getValue().getClass());**

**else System.out.println("HashMap'ът е празен!");**

**System.out.println("\n");**

**}**

**});**

**ctrlPanel.add(btnHashMapToCons);**

**lNum.setBounds(5, 550, 95, 20);**

**ctrlPanel.add(lNum);**

**numPr.setBounds(100, 550, 30, 20);**

**ctrlPanel.add(numPr);**

**x1.setBounds(100, 150, 80, 20);**

**y1.setBounds(100, 180, 80, 20);**

**x2.setBounds(100, 210, 80, 20);**

**y2.setBounds(100, 240, 80, 20);**

**Label l1 = new Label("X1:");**

**l1.setBounds(75, 150, 20, 20);**

**ctrlPanel.add(l1);**

**ctrlPanel.add(x1);**

**Label l2 = new Label("Y1:");**

**l2.setBounds(75, 180, 20, 20);**

**ctrlPanel.add(l2);**

**ctrlPanel.add(y1);**

**Label l3 = new Label("X2:");**

**l3.setBounds(75, 210, 20, 20);**

**ctrlPanel.add(l3);**

**ctrlPanel.add(x2);**

**Label l4 = new Label("Y2:");**

**l4.setBounds(75, 240, 20, 20);**

**ctrlPanel.add(l4);**

**ctrlPanel.add(y2);**

**drawPanel.setLayout(null);**

**drawPanel.setSize(793, 597);**

**paintPanel.setBounds(5, 5, 783, 560);**

**paintPanel.setBackground(curBGCol);**

**drawPanel.add(paintPanel);**

**f.add(ctrlPanel);**

**f.add(drawPanel);**

**f.setVisible(true);**

**}**

**/\* Функция за извеждане на съобщение във фрайм \*/**

**void messageFrame(String msg) {**

**Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();**

**final Frame msgFrame = new Frame("Съобщение");**

**msgFrame.setLocation((screenSize.width - 200) / 2, (screenSize.height - 120) / 2);**

**msgFrame.setResizable(false);**

**msgFrame.addWindowListener(new WindowAdapter () {**

**public void windowClosing(WindowEvent e) {**

**msgFrame.setVisible(false);**

**}**

**});**

**msgFrame.setVisible(true);**

**msgFrame.setLayout(new GridLayout(2,1));**

**msgFrame.add(new Label(msg));**

**Button btnOk = new Button("OK");**

**btnOk.setSize(100, 25);**

**btnOk.addActionListener(new ActionListener() {**

**public void actionPerformed(ActionEvent e) {**

**if (msgB > -1)**

**msgFrame.setVisible(false);**

**else**

**System.exit(0);**

**}**

**});**

**msgFrame.add(btnOk);**

**msgFrame.pack();**

**}**

**/\* MAIN \*/**

**public static void main(String[] args) {**

**new Main();**

**}**

**}**

* **ImagePanel.java**

**package mypack;**

**import java.awt.Canvas;**

**import java.awt.Color;**

**import java.awt.Graphics;**

**import java.awt.image.BufferedImage;**

**import java.io.File;**

**import java.io.IOException;**

**import javax.imageio.ImageIO;**

**class ImagePanel extends Canvas {**

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

**BufferedImage imageCol;**

**Color fillCol1, fillCol2,**

**fillCol3, borderCol;**

**static int br;**

**ImagePanel() {**

**br = 0;**

**fillCol1 = fillCol2 = fillCol3 = Color.black;**

**File file = new File("colors.jpg");**

**try {**

**imageCol = ImageIO.read(file);**

**} catch (IOException e1) {**

**new Main(-1).messageFrame("Изображението с цветовете не е намерен!");**

**}**

**}**

**public void paint(Graphics g) {**

**g.setColor(Color.black);**

**g.drawString("Цвят:", 5, 16);**

**if (br == 0)**

**borderCol = Color.red;**

**else borderCol = Color.white;**

**g.setColor(borderCol);**

**g.drawRect(50, 2, 20, 20);**

**g.setColor(fillCol1);**

**g.fillRect(51, 3, 19, 19);**

**if (br == 1)**

**borderCol = Color.red;**

**else borderCol = Color.white;**

**g.setColor(borderCol);**

**g.drawRect(80, 2, 20, 20);**

**g.setColor(fillCol2);**

**g.fillRect(81, 3, 19, 19);**

**if (br == 2)**

**borderCol = Color.red;**

**else borderCol = Color.white;**

**g.setColor(borderCol);**

**g.drawRect(110, 2, 20, 20);**

**g.setColor(fillCol3);**

**g.fillRect(111, 3, 19, 19);**

**g.drawImage(imageCol, 0, 25, null);**

**}**

**}**

* **PaintPanel.java**

**package mypack;**

**import java.awt.Canvas;**

**import java.awt.Graphics;**

**import java.util.HashMap;**

**import java.util.Map;**

**public class PaintPanel extends Canvas {**

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

**HashMap<String, GraphicObject> hm1;**

**PaintPanel(HashMap<String, GraphicObject> hm) {**

**hm1 = hm;**

**}**

**public void paint(Graphics g) {**

**for (Map.Entry<String, GraphicObject> me : hm1.entrySet())**

**me.getValue().draw(g);**

**}**

**}**

* **GraphicObject.java**

**package mypack;**

**import java.awt.Graphics;**

**public interface GraphicObject {**

**public abstract void draw(Graphics g);**

**}**

* **NotFilledRectangle.java**

**package mypack;**

**import java.awt.Color;**

**import java.awt.Graphics;**

**public class NotFilledRectangle implements GraphicObject {**

**public int x, y, w, h;**

**public Color drawCol;**

**NotFilledRectangle(int x, int y, int w, int h, Color col) {**

**this.x = x; this.y = y;**

**this.w = w; this.h = h;**

**drawCol = col;**

**}**

**public void draw(Graphics g) {**

**g.setColor(drawCol);**

**g.drawRect(x, y, w, h);**

**}**

**}**

* **FilledRectangle.java**

**package mypack;**

**import java.awt.Color;**

**import java.awt.Graphics;**

**public class FilledRectangle implements GraphicObject {**

**public int x, y, w, h;**

**public Color drawCol, fillCol;**

**FilledRectangle(int x, int y, int w, int h, Color col, Color fCol) {**

**this.x = x; this.y = y;**

**this.w = w;this.h = h;**

**drawCol = col; fillCol = fCol;**

**}**

**public void draw(Graphics g) {**

**g.setColor(drawCol);**

**g.drawRect(x, y, w, h);**

**g.setColor(fillCol);**

**g.fillRect(x+1, y+1, w-1, h-1);**

**}**

**}**

* **FilledOval.java**

package mypack;

import java.awt.Color;

import java.awt.Graphics;

public class FilledOval implements GraphicObject {

public int x, y, w, h;

public Color drawCol, fillCol;

FilledOval(int x, int y, int w, int h, Color col, Color fCol) {

this.x = x; this.y = y;

this.w = w;this.h = h;

drawCol = col; fillCol = fCol;

}

public void draw(Graphics g) {

g.setColor(drawCol);

g.drawOval(x, y, w, h);

g.setColor(fillCol);

g.fillOval(x+1, y+1, w-1, h-1);

}

}