**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение высшего образования  
 «Кемеровский государственный университет»

Институт фундаментальных наук

Кафедра ЮНЕСКО по информационным вычислительным технологиям

**ОТЧЕТ**

по учебной практике, технологической (проектно-технологической) практике

проект «свой paint с нуля»

(название проекта)

студента 1 курса

Хроменок Екатерина Андреевна

(ФИО полностью)

направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем.

направленность (профиль) подготовки «Информационные системы и базы данных».

Руководитель практики:

канд. физ.-мат. наук, доцент

К.С. Иванов

(ученая степень, звание, должность, ФИО)

Зав. кафедрой ЮНЕСКО по ИВТ

доктор физ.-мат. наук, профессор

Ю.Н. Захаров

(ученая степень, звание, должность, ФИО)

Работа защищена с оценкой:

(оценка)

« » 2021 г.

Кемерово 2021

Оглавление

[О проекте 2](#_Toc85049856)

[Актуальность 2](#_Toc85049857)

[Теоритическая значимость 2](#_Toc85049858)

[Практическая значимость 2](#_Toc85049859)

[Состав проекта 2](#_Toc85049860)

[Цель 2](#_Toc85049861)

[Задача 3](#_Toc85049862)

[План-график проекта 3](#_Toc85049863)

[Что такое Java 3](#_Toc85049864)

[Что такое Paint 4](#_Toc85049865)

[Используемые программные средства 4](#_Toc85049866)

[Что представляет собой Java. 4](#_Toc85049867)

[Ход работы 5](#_Toc85049868)

[12.03.21-26.03.21 5](#_Toc85049869)

[27.03.21-15.04.21 5](#_Toc85049870)

[16.04.21-28.04.21 5](#_Toc85049871)

[29.04.21-14.05.21 15](#_Toc85049872)

[Вывод 26](#_Toc85049875)

[Литература 27](#_Toc85049876)

# О проекте

Изучить азы Java. Воспользовавшись изученным, написать код и создать собственное приложение.

# Актуальность

На сегодняшний день во многих профессиях постоянно возникает необходимость использования [графических программ](http://www.logos34.ru/obuchenie/raspisanie_kursov/kompyuternye/kompyuternyj_dizajn_i_grafika/). Причем к данным программам обращаются не только дизайнеры, художники, фотографы, полиграфисты, редакторы изданий, веб-мастера, но и многие другие специалисты, которым приходится быстро и качественно обрабатывать какие-то изображения.

# Теоритическая значимость

Теоритическая значимость моей работы заключается в том, что я на основании изучения «Java», выявила, что писать коды и создавать необычные приложения не сложно и реалистично, даже, если вы впервые столкнулись с программированием.

# Практическая значимость

Материал данного проекта, на мой взгляд, может быть использован теми, кому интересны рисование и программирование одновременно.

# Состав проекта

Хроменок Екатерина, группы МОА-205, Логин на github.com: 96AmazinG

# Цель

Составить презентацию и отчет о проделанной работе. Познакомить студентов с основными возможностями Java и приложения

# Задача

Узнать и изучить способы создания и написания кода на Java для собственного приложения.

# План-график проекта

12.03.21-26.03.21 - планирование и обсуждение понравившейся темы.

27.03.21-15.04.21 - изучение теоретического материала.

16.04.21-28.04.21 - практиковать изученный материал.

29.04.21-14.05.21 - создания конечной работы, составление отчёта и редактирование до полного идеализма.

# Что такое Java

Java - это технология, используемая для разработки приложений, которые делают работу в сети Интернет более увлекательной и удобной. Java отличается от языка программирования javascript, который представляет собой простую технологию для создания веб-страниц и выполняется только в браузере.

С помощью Java вы можете играть в игры, загружать фотографии, общаться в режиме онлайн, совершать виртуальные экскурсии и пользоваться такими услугами как дистанционное обучение, дистанционное банковское обслуживание и просмотр интерактивных карт. Если у вас не установлена среда Java, многие приложения и веб-сайты просто не будут работать.

По умолчанию Java автоматически уведомляет вас о готовых к установке новых обновлениях. Чтобы всегда быть в курсе изменений и поддерживать безопасность компьютера, необходимо принять и установить эти обновления. Если вы получаете уведомление об обновлении версии Java на компьютере под управлением Windows, но никогда не загружали и не устанавливали обновления, вполне вероятно, что среда Java уже предварительно установлена на вашем новом компьютере.

# Что такое Paint

Microsoft Paint – достаточно простой [графический редактор.](https://yandex.ru/turbo/syl.ru/s/article/77034/vektornyiy-i-rastrovyiy-graficheskie-redaktoryi?parent-reqid=1624216373789764-1712190502058609512521648-production-app-host-vla-web-yp-259&utm_source=turbo_turbo) Является одной из обязательных программ. Идет в комплекте с операционной системой [Microsoft Windows.](https://yandex.ru/turbo/syl.ru/s/article/387875/microsoft-windows-istoriya-sozdaniya-i-interesnyie-faktyi?parent-reqid=1624216373789764-1712190502058609512521648-production-app-host-vla-web-yp-259&utm_source=turbo_turbo) Графический редактор Paint предназначен для создания и редактирования рисунков различной цветовой гаммы и чертежей. Выполненные в этом редакторе работы называются растровыми.

Под растровой графикой подразумевается совокупность строк, состоящих из пикселей. Их набор образует двумерный массив, именуемый растр. Стоит отметить, что у данного редактора есть один важный минус – плохое масштабирование. Подобное построение приводит к тому, что при увеличении или уменьшении размера файла происходит искажение рисунка.

Графический интерфейс пользователя является частью пользовательского интерфейса и определяет взаимодействие с пользователем на уровне визуализированной информации.

# Используемые программные средства

1. Microsoft Word
2. Github
3. Java
4. [NetBeans](http://netbeans.org/downloads/)

# Что представляет собой Java.

1) Java представляет собой язык программирования и платформу вычислений, которая была впервые выпущена Sun Microsystems в 1995 г. Существует множество приложений и веб-сайтов, которые не работают при отсутствии установленной Java, и с каждым днем число таких веб-сайтов и приложений увеличивается. Java отличается быстротой, высоким уровнем защиты и надежностью. От портативных компьютеров до центров данных, от игровых консолей до суперкомпьютеров, используемых для научных разработок, от сотовых телефонов до сети Интернет — Java повсюду!

2) Программное обеспечение подключаемого модуля Java является компонентом Java Runtime Environment (JRE). JRE обеспечивает возможность запуска апплетов, написанных на языке программирования Java, в различных браузерах. Подключаемый модуль Java не является автономной программой и не может быть установлен отдельно.

# Ход работы

# 12.03.21-26.03.21

Выбрали более интересную тему для изучения. Загрузили Java на компьютер, ознакомились с интерфейсом, синтаксисом набора команд и структурой приложения. Так же нашли дидактический материал по выбранной теме.

# 27.03.21-15.04.21

Нашли дидактический материал по выбранной теме. Перечитали несколько статей в интернете. Изучили набор команд, которые понадобятся для реализации выбранной темы.

# 16.04.21-28.04.21

Мы приступили к практической части.

Режим рисования

int rezhim=0;

int xPad;

int xf;

int yf;

int yPad;

int thickness;

boolean pressed=false;

текущий цвет

Color maincolor;

MyFrame f;

MyPanel japan;

JButton colorbutton;

JColorChooser tcc;

поверхность рисования

BufferedImage imag;

если мы загружаем картинку

boolean loading=false;

String fileName;

public ImageEdit()

{

f=new MyFrame("мой пейнт");

f.setSize(350,350);

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

maincolor=Color.black;

JMenuBar menuBar = new JMenuBar();

f.setJMenuBar(menuBar);

menuBar.setBounds(0,0,350,30);

JMenu fileMenu = new JMenu("Файл");

menuBar.add(fileMenu);

Action loadAction = new AbstractAction("Загрузить")

{

public void actionPerformed(ActionEvent event)

{

JFileChooser jf= new JFileChooser();

int result = jf.showOpenDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

try

{

fileName = jf.getSelectedFile().getAbsolutePath();

File iF= new File(fileName);

jf.addChoosableFileFilter(new TextFileFilter(".png"));

jf.addChoosableFileFilter(new TextFileFilter(".jpg"));

imag = ImageIO.read(iF);

loading=true;

f.setSize(imag.getWidth()+40, imag.getWidth()+80);

japan.setSize(imag.getWidth(), imag.getWidth());

japan.repaint();

} catch (FileNotFoundException ex) {

JOptionPane.showMessageDialog(f, "Такого файла не существует");

}

catch (IOException ex) {

JOptionPane.showMessageDialog(f, "Исключение ввода-вывода");

}

catch (Exception ex) {

}

}

}

};

JMenuItem loadMenu = new JMenuItem(loadAction);

fileMenu.add(loadMenu);

Action saveAction = new AbstractAction("Сохранить")

{

public void actionPerformed(ActionEvent event)

{

try

{

JFileChooser jf= new JFileChooser();

Создаем фильтры файлов

TextFileFilter pngFilter = new TextFileFilter(".png");

TextFileFilter jpgFilter = new TextFileFilter(".jpg");

if(fileName==null)

{

Добавляем фильтры

jf.addChoosableFileFilter(pngFilter);

jf.addChoosableFileFilter(jpgFilter);

int result = jf.showSaveDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

fileName = jf.getSelectedFile().getAbsolutePath();

}

}

Смотрим какой фильтр выбран

if(jf.getFileFilter()==pngFilter)

{

ImageIO.write(imag, "png", new File(fileName+".png"));

}

else

{

ImageIO.write(imag, "jpeg", new File(fileName+".jpg"));

}

}

catch(IOException ex)

{

JOptionPane.showMessageDialog(f, "Ошибка ввода-вывода");

}

}

};

JMenuItem saveMenu = new JMenuItem(saveAction);

fileMenu.add(saveMenu);

Action saveasAction = new AbstractAction("Сохранить как...")

{

public void actionPerformed(ActionEvent event)

{

try

{

JFileChooser jf= new JFileChooser();

Создаем фильтры для файлов

TextFileFilter pngFilter = new TextFileFilter(".png");

TextFileFilter jpgFilter = new TextFileFilter(".jpg");

Добавляем фильтры

jf.addChoosableFileFilter(pngFilter);

jf.addChoosableFileFilter(jpgFilter);

int result = jf.showSaveDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

fileName = jf.getSelectedFile().getAbsolutePath();

}

элементы управления для кнопок

JToolBar colorbar = new JToolBar("Colorbar", JToolBar.HORIZONTAL);

colorbar.setBounds(30, 0, 260, 30);

colorbutton = new JButton();

colorbutton.setBackground(maincolor);

colorbutton.setBounds(15, 5, 20, 20);

colorbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

ColorDialog coldi = new ColorDialog(f,"Выбор цвета");

coldi.setVisible(true);

}

});

colorbar.add(colorbutton);

JButton redbutton = new JButton();

redbutton.setBackground(Color.red);

redbutton.setBounds(40, 5, 15, 15);

redbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.red;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(redbutton);

JButton orangebutton = new JButton();

orangebutton.setBackground(Color.orange);

orangebutton.setBounds(60, 5, 15, 15);

orangebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.orange;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(orangebutton);

JButton yellowbutton = new JButton();

yellowbutton.setBackground(Color.yellow);

yellowbutton.setBounds(80, 5, 15, 15);

yellowbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.yellow;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(yellowbutton);

JButton greenbutton = new JButton();

greenbutton.setBackground(Color.green);

greenbutton.setBounds(100, 5, 15, 15);

greenbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.green;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(greenbutton);

JButton bluebutton = new JButton();

bluebutton.setBackground(Color.blue);

bluebutton.setBounds(120, 5, 15, 15);

bluebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.blue;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(bluebutton);

JButton cyanbutton = new JButton();

cyanbutton.setBackground(Color.cyan);

cyanbutton.setBounds(140, 5, 15, 15);

cyanbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.cyan;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(cyanbutton);

JButton magentabutton = new JButton();

magentabutton.setBackground(Color.magenta);

magentabutton.setBounds(160, 5, 15, 15);

magentabutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.magenta;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(magentabutton);

JButton whitebutton = new JButton();

whitebutton.setBackground(Color.white);

whitebutton.setBounds(180, 5, 15, 15);

whitebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.white;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(whitebutton);

JButton blackbutton = new JButton();

blackbutton.setBackground(Color.black);

blackbutton.setBounds(200, 5, 15, 15);

blackbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.black;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(blackbutton);

colorbar.setLayout(null);

f.add(colorbar);

tcc = new JColorChooser(maincolor);

tcc.getSelectionModel().addChangeListener(new ChangeListener()

{

public void stateChanged(ChangeEvent e)

{

maincolor = tcc.getColor();

colorbutton.setBackground(maincolor);

}

});

japan.addMouseMotionListener(new MouseMotionAdapter()

{

public void mouseDragged(MouseEvent e)

{

if (pressed==true)

{

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

установка цвета

g2.setColor(maincolor);

switch (rezhim)

{

карандаш

case 0:

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

кисть

case 1:

g2.setStroke(new BasicStroke(3.0f));

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

ластик

case 2:

g2.setStroke(new BasicStroke(3.0f));

g2.setColor(Color.WHITE);

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

}

xPad=e.getX();

yPad=e.getY();

}

japan.repaint();

}

});

текст

case 3:

// устанавливаем фокус для панели,

// чтобы печатать на ней текст

japan.requestFocus();

break;

}

xPad=e.getX();

yPad=e.getY();

pressed=true;

japan.repaint();

}

public void mousePressed(MouseEvent e) {

xPad=e.getX();

yPad=e.getY();

xf=e.getX();

yf=e.getY();

pressed=true;

}

public void mouseReleased(MouseEvent e) {

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

g2.setColor(maincolor);

расчеты для овала и прямоугольника

int x1=xf, x2=xPad, y1=yf, y2=yPad;

if(xf>xPad)

{

x2=xf; x1=xPad;

}

if(yf>yPad)

{

y2=yf; y1=yPad;

}

switch(rezhim)

{

Линия

case 4:

g.drawLine(xf, yf, e.getX(), e.getY());

break;

круг

case 5:

g.drawOval(x1, y1, (x2-x1), (y2-y1));

break;

// прямоугольник

case 6:

g.drawRect(x1, y1, (x2-x1), (y2-y1));

break;

}

xf=0; yf=0;

pressed=false;

japan.repaint();

}

});

japan.addKeyListener(new KeyAdapter()

{

public void keyReleased(KeyEvent e)

{

устанавливаем фокус для панели, чтобы печатать текст

japan.requestFocus();

}

public void keyTyped(KeyEvent e)

{

if(rezhim==3){

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

установка цвета

g2.setColor(maincolor);

g2.setStroke(new BasicStroke(2.0f));

String str = new String("");

str+=e.getKeyChar();

g2.setFont(new Font("Arial", 0, 15));

g2.drawString(str, xPad, yPad);

xPad+=10;

f.addComponentListener(new ComponentAdapter() {

public void componentResized(java.awt.event.ComponentEvent evt) {

если делаем загрузку, то изменение размеров формы, отрабатываем в коде загрузки

if(loading==false)

{

japan.setSize(f.getWidth()-40, f.getHeight()-80);

BufferedImage tempImage = new BufferedImage(japan.getWidth(), japan.getHeight(), BufferedImage.TYPE\_INT\_RGB);

Graphics2D d2 = (Graphics2D) tempImage.createGraphics();

d2.setColor(Color.white);

d2.fillRect(0, 0, japan.getWidth(), japan.getHeight());

tempImage.setData(imag.getRaster());

imag=tempImage;

japan.repaint();

}

loading=false;

}

});

f.setLayout(null);

f.setVisible(true);

}

public static void main(String[] args) {

SwingUtilities.invokeLater(new Runnable() {

public void run() {

new ImageEdit();

}

});

}

class ColorDialog extends JDialog

{

public ColorDialog(JFrame owner, String title)

{

super(owner, title, true);

add(tcc);

setSize(200, 200);

}

}

class MyFrame extends JFrame

{

public void paint(Graphics g)

{

super.paint(g);

}

public MyFrame(String title)

{

super(title);

}

}

class MyPanel extends JPanel

{

public MyPanel()

{ }

public void paintComponent (Graphics g)

{

if(imag==null)

{

imag = new BufferedImage(this.getWidth(), this.getHeight(), BufferedImage.TYPE\_INT\_RGB);

Graphics2D d2 = (Graphics2D) imag.createGraphics();

d2.setColor(Color.white);

d2.fillRect(0, 0, this.getWidth(), this.getHeight());

}

super.paintComponent(g);

g.drawImage(imag, 0, 0,this);

}

}

Фильтр картинок

class TextFileFilter extends FileFilter

{

private String ext;

public TextFileFilter(String ext)

{

this.ext=ext;

}

public boolean accept(java.io.File file)

{

if (file.isDirectory()) return true;

return (file.getName().endsWith(ext));

}

public String getDescription()

{

return "\*"+ext;

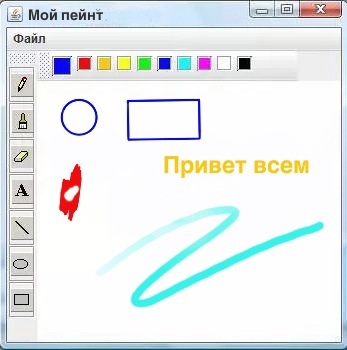
}

}

}

29.04.21-14.05.21

В результате изучения некоторых команд. Мы создали собственное приложение. В нём есть самые минимальный, но необходимый функционал: карандаш, кисть, ластик, выбор цвета и рисование простейших фигур. Кроме того можно будет загружать изображения и редактировать их и сохранять. Работа с изображениями в формате png.



Конечный код

import java.awt.\*;

import java.awt.geom.\*;

import java.awt.event.\*;

import java.io.\*;

import javax.swing.\*;

import javax.swing.event.\*;

import java.awt.image.\*;

import javax.imageio.\*;

import javax.swing.filechooser.FileFilter;

public class ImageEdit

{

int rezhim=0;

int xPad;

int xf;

int yf;

int yPad;

int thickness;

boolean pressed=false;

Color maincolor;

MyFrame f;

MyPanel japan;

JButton colorbutton;

JColorChooser tcc;

BufferedImage imag;

boolean loading=false;

String fileName;

public ImageEdit()

{

f=new MyFrame("Графический редактор");

f.setSize(350,350);

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

maincolor=Color.black;

JMenuBar menuBar = new JMenuBar();

f.setJMenuBar(menuBar);

menuBar.setBounds(0,0,350,30);

JMenu fileMenu = new JMenu("Файл");

menuBar.add(fileMenu);

Action loadAction = new AbstractAction("Загрузить")

{

public void actionPerformed(ActionEvent event)

{

JFileChooser jf= new JFileChooser();

int result = jf.showOpenDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

try

{

fileName = jf.getSelectedFile().getAbsolutePath();

File iF= new File(fileName);

jf.addChoosableFileFilter(new TextFileFilter(".png"));

jf.addChoosableFileFilter(new TextFileFilter(".jpg"));

imag = ImageIO.read(iF);

loading=true;

f.setSize(imag.getWidth()+40, imag.getWidth()+80);

japan.setSize(imag.getWidth(), imag.getWidth());

japan.repaint();

} catch (FileNotFoundException ex) {

JOptionPane.showMessageDialog(f, "Такого файла не существует");

}

catch (IOException ex) {

JOptionPane.showMessageDialog(f, "Исключение ввода-вывода");

}

catch (Exception ex) {

}

}

}

};

JMenuItem loadMenu = new JMenuItem(loadAction);

fileMenu.add(loadMenu);

Action saveAction = new AbstractAction("Сохранить")

{

public void actionPerformed(ActionEvent event)

{

try

{

JFileChooser jf= new JFileChooser();

TextFileFilter pngFilter = new TextFileFilter(".png");

TextFileFilter jpgFilter = new TextFileFilter(".jpg");

if(fileName==null)

{

jf.addChoosableFileFilter(pngFilter);

jf.addChoosableFileFilter(jpgFilter);

int result = jf.showSaveDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

fileName = jf.getSelectedFile().getAbsolutePath();

}

}

if(jf.getFileFilter()==pngFilter)

{

ImageIO.write(imag, "png", new File(fileName+".png"));

}

else

{

ImageIO.write(imag, "jpeg", new File(fileName+".jpg"));

}

}

catch(IOException ex)

{

JOptionPane.showMessageDialog(f, "Ошибка ввода-вывода");

}

}

};

JMenuItem saveMenu = new JMenuItem(saveAction);

fileMenu.add(saveMenu);

Action saveasAction = new AbstractAction("Сохранить как...")

{

public void actionPerformed(ActionEvent event)

{

try

{

JFileChooser jf= new JFileChooser();

TextFileFilter pngFilter = new TextFileFilter(".png");

TextFileFilter jpgFilter = new TextFileFilter(".jpg");

jf.addChoosableFileFilter(pngFilter);

jf.addChoosableFileFilter(jpgFilter);

int result = jf.showSaveDialog(null);

if(result==JFileChooser.APPROVE\_OPTION)

{

fileName = jf.getSelectedFile().getAbsolutePath();

}

if(jf.getFileFilter()==pngFilter)

{

ImageIO.write(imag, "png", new File(fileName+".png"));

}

else

{

ImageIO.write(imag, "jpeg", new File(fileName+".jpg"));

}

}

catch(IOException ex)

{

JOptionPane.showMessageDialog(f, "Ошибка ввода-вывода");

}

}

};

JMenuItem saveasMenu = new JMenuItem(saveasAction);

fileMenu.add(saveasMenu);

japan = new MyPanel();

japan.setBounds(30,30,260,260);

japan.setBackground(Color.white);

japan.setOpaque(true);

f.add(japan);

JToolBar toolbar = new JToolBar("Toolbar", JToolBar.VERTICAL);

JButton penbutton = new JButton(new ImageIcon("pen.png"));

penbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=0;

}

});

toolbar.add(penbutton);

JButton brushbutton = new JButton(new ImageIcon("brush.png"));

brushbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=1;

}

});

toolbar.add(brushbutton);

JButton lasticbutton = new JButton(new ImageIcon("lastic.png"));

lasticbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=2;

}

});

toolbar.add(lasticbutton);

JButton textbutton = new JButton(new ImageIcon("text.png"));

textbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=3;

}

});

toolbar.add(textbutton);

JButton linebutton = new JButton(new ImageIcon("line.png"));

linebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=4;

}

});

toolbar.add(linebutton);

JButton elipsbutton = new JButton(new ImageIcon("elips.png"));

elipsbutton.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent event)

{

rezhim=5;

}

});

toolbar.add(elipsbutton);

JButton rectbutton = new JButton(new ImageIcon("rect.png"));

rectbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

rezhim=6;

}

});

toolbar.add(rectbutton);

toolbar.setBounds(0, 0, 30, 300);

f.add(toolbar);

JToolBar colorbar = new JToolBar("Colorbar", JToolBar.HORIZONTAL);

colorbar.setBounds(30, 0, 260, 30);

colorbutton = new JButton();

colorbutton.setBackground(maincolor);

colorbutton.setBounds(15, 5, 20, 20);

colorbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

ColorDialog coldi = new ColorDialog(f,"Выбор цвета");

coldi.setVisible(true);

}

});

colorbar.add(colorbutton);

JButton redbutton = new JButton();

redbutton.setBackground(Color.red);

redbutton.setBounds(40, 5, 15, 15);

redbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.red;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(redbutton);

JButton orangebutton = new JButton();

orangebutton.setBackground(Color.orange);

orangebutton.setBounds(60, 5, 15, 15);

orangebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.orange;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(orangebutton);

JButton yellowbutton = new JButton();

yellowbutton.setBackground(Color.yellow);

yellowbutton.setBounds(80, 5, 15, 15);

yellowbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.yellow;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(yellowbutton);

JButton greenbutton = new JButton();

greenbutton.setBackground(Color.green);

greenbutton.setBounds(100, 5, 15, 15);

greenbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.green;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(greenbutton);

JButton bluebutton = new JButton();

bluebutton.setBackground(Color.blue);

bluebutton.setBounds(120, 5, 15, 15);

bluebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.blue;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(bluebutton);

JButton cyanbutton = new JButton();

cyanbutton.setBackground(Color.cyan);

cyanbutton.setBounds(140, 5, 15, 15);

cyanbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.cyan;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(cyanbutton);

JButton magentabutton = new JButton();

magentabutton.setBackground(Color.magenta);

magentabutton.setBounds(160, 5, 15, 15);

magentabutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.magenta;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(magentabutton);

JButton whitebutton = new JButton();

whitebutton.setBackground(Color.white);

whitebutton.setBounds(180, 5, 15, 15);

whitebutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.white;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(whitebutton);

JButton blackbutton = new JButton();

blackbutton.setBackground(Color.black);

blackbutton.setBounds(200, 5, 15, 15);

blackbutton.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

maincolor = Color.black;

colorbutton.setBackground(maincolor);

}

});

colorbar.add(blackbutton);

colorbar.setLayout(null);

f.add(colorbar);

tcc = new JColorChooser(maincolor);

tcc.getSelectionModel().addChangeListener(new ChangeListener()

{

public void stateChanged(ChangeEvent e)

{

maincolor = tcc.getColor();

colorbutton.setBackground(maincolor);

}

});

japan.addMouseMotionListener(new MouseMotionAdapter()

{

public void mouseDragged(MouseEvent e)

{

if (pressed==true)

{

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

g2.setColor(maincolor);

switch (rezhim)

{

case 0:

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

case 1:

g2.setStroke(new BasicStroke(3.0f));

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

case 2:

g2.setStroke(new BasicStroke(3.0f));

g2.setColor(Color.WHITE);

g2.drawLine(xPad, yPad, e.getX(), e.getY());

break;

}

xPad=e.getX();

yPad=e.getY();

}

japan.repaint();

}

});

japan.addMouseListener(new MouseAdapter()

{

public void mouseClicked(MouseEvent e) {

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

g2.setColor(maincolor);

switch (rezhim)

{

case 0:

g2.drawLine(xPad, yPad, xPad+1, yPad+1);

break;

case 1:

g2.setStroke(new BasicStroke(3.0f));

g2.drawLine(xPad, yPad, xPad+1, yPad+1);

break;

case 2:

g2.setStroke(new BasicStroke(3.0f));

g2.setColor(Color.WHITE);

g2.drawLine(xPad, yPad, xPad+1, yPad+1);

break;

case 3:

japan.requestFocus();

break;

}

xPad=e.getX();

yPad=e.getY();

pressed=true;

japan.repaint();

}

public void mousePressed(MouseEvent e) {

xPad=e.getX();

yPad=e.getY();

xf=e.getX();

yf=e.getY();

pressed=true;

}

public void mouseReleased(MouseEvent e) {

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

g2.setColor(maincolor);

int x1=xf, x2=xPad, y1=yf, y2=yPad;

if(xf>xPad)

{

x2=xf; x1=xPad;

}

if(yf>yPad)

{

y2=yf; y1=yPad;

}

switch(rezhim)

{

case 4:

g.drawLine(xf, yf, e.getX(), e.getY());

break;

case 5:

g.drawOval(x1, y1, (x2-x1), (y2-y1));

break;

case 6:

g.drawRect(x1, y1, (x2-x1), (y2-y1));

break;

}

xf=0; yf=0;

pressed=false;

japan.repaint();

}

});

japan.addKeyListener(new KeyAdapter()

{

public void keyReleased(KeyEvent e)

{

japan.requestFocus();

}

public void keyTyped(KeyEvent e)

{

if(rezhim==3){

Graphics g = imag.getGraphics();

Graphics2D g2 = (Graphics2D)g;

// установка цвета

g2.setColor(maincolor);

g2.setStroke(new BasicStroke(2.0f));

String str = new String("");

str+=e.getKeyChar();

g2.setFont(new Font("Arial", 0, 15));

g2.drawString(str, xPad, yPad);

xPad+=10;

japan.requestFocus();

japan.repaint();

}

}

});

f.addComponentListener(new ComponentAdapter() {

public void componentResized(java.awt.event.ComponentEvent evt) {

if(loading==false)

{

japan.setSize(f.getWidth()-40, f.getHeight()-80);

BufferedImage tempImage = new BufferedImage(japan.getWidth(), japan.getHeight(), BufferedImage.TYPE\_INT\_RGB);

Graphics2D d2 = (Graphics2D) tempImage.createGraphics();

d2.setColor(Color.white);

d2.fillRect(0, 0, japan.getWidth(), japan.getHeight());

tempImage.setData(imag.getRaster());

imag=tempImage;

japan.repaint();

}

loading=false;

}

});

f.setLayout(null);

f.setVisible(true);

}

public static void main(String[] args) {

SwingUtilities.invokeLater(new Runnable() {

public void run() {

new ImageEdit();

}

});

}

class ColorDialog extends JDialog

{

public ColorDialog(JFrame owner, String title)

{

super(owner, title, true);

add(tcc);

setSize(200, 200);

}

}

class MyFrame extends JFrame

{

public void paint(Graphics g)

{

super.paint(g);

}

public MyFrame(String title)

{

super(title);

}

}

class MyPanel extends JPanel

{

public MyPanel()

{ }

public void paintComponent (Graphics g)

{

if(imag==null)

{

imag = new BufferedImage(this.getWidth(), this.getHeight(), BufferedImage.TYPE\_INT\_RGB);

Graphics2D d2 = (Graphics2D) imag.createGraphics();

d2.setColor(Color.white);

d2.fillRect(0, 0, this.getWidth(), this.getHeight());

}

super.paintComponent(g);

g.drawImage(imag, 0, 0,this);

}

}

class TextFileFilter extends FileFilter

{

private String ext;

public TextFileFilter(String ext)

{

this.ext=ext;

}

public boolean accept(java.io.File file)

{

if (file.isDirectory()) return true;

return (file.getName().endsWith(ext));

}

public String getDescription()

{

return "\*"+ext;

}

}

}

Вывод

В процессе работы над проектом мы ознакомились с Java с некоторыми его командами и реализовали собственное приложение. В своей работе мы ставили задачу: Узнать и изучить способы создания и написания кода на Java для приложения. Нам удалось создать условия для творческой самореализации.

Литература

1. Фримен Э., Робсон Э., Сьерра К., Бейтс Б.Х99 Head First. Паттерны проектирования. Обновленное юбилейное издание. — СПб.: Питер, 2018. — 656 с.: ил. — (Серия «Head First O’Reilly»).
2. Хорстманн, Кей С.Х82 Java. Библиотека профессионала, том 1. Основы. 10-е изд.: Пер. с англ. — М .: ООО "И.Д. Вильямс", 2016. — 864 с .: ил. — Парал. тит. англ.
3. Шилдт, Ге рберт. Ш57 Java 8. Полное руководство; 9-е изд.: Пер. с англ. -М. : ООО "И.Д. Вильяме", 2015. - 1376 с. : ил. - Парал. тит. англ.