package 一个绘图APP;

import java.awt.BasicStroke;

import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.Container;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.Shape;

import java.awt.event.\*;

import java.awt.geom.Line2D;

import java.awt.geom.Point2D;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.util.LinkedList;

import java.util.List;

import javax.swing.\*;

import javax.swing.border.\*;

import javax.swing.filechooser.FileSystemView;

public class Mouse extends JFrame implements MouseListener{

public JPanel p;

public JPanel t;

boolean te=false;

JMenuBar menubar;

Color a=new Color(0,0,0);

int x1,y1;

int x=0,y;

File file;

private List<MyLine>lineList=new LinkedList<>();

MyLine myline;

Mouse()

{

init();

}

public void init()

{

file=new File("Shape.shuai");

p = new JPanel();

//注册鼠标监听

p.addMouseListener(this);

setTitle("绘图APP");

//目录栏

JMenuBar menubar=new JMenuBar();

JMenu menu1=new JMenu("File");

JMenu menu2=new JMenu("Color");

//换画笔颜色

JMenuItem item1=new JMenuItem("red");

item1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK\_S,ActionEvent.CTRL\_MASK));

item1.addActionListener(e->{

a=new Color(250,0,0);

});

JMenuItem item2=new JMenuItem("black");

item2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK\_G,ActionEvent.CTRL\_MASK));

item2.addActionListener(e->{

a=new Color(0,0,0);

});

JMenuItem item3=new JMenuItem("blue");

item3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK\_T,ActionEvent.CTRL\_MASK));

item3.addActionListener(e->{

a=new Color(0,0,250);

});

//文件打开

JMenuItem item4=new JMenuItem("open");

item4.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK\_H,ActionEvent.CTRL\_MASK));

item4.addActionListener(e->{

try {

FileInputStream fileIn = new FileInputStream(file);

ObjectInputStream objectIn = new ObjectInputStream(fileIn);

Graphics2D g2 = (Graphics2D)getGraphics();

//设置画笔大小

BasicStroke bs\_1=new BasicStroke(3,BasicStroke.CAP\_BUTT,BasicStroke.JOIN\_BEVEL);

g2.setStroke(bs\_1);

Color color = g2.getColor();

g2.setColor(Color.black);

/\*for(MyLine myline:lineList) {

g2.draw(myline.line);

}\*/

Object obj = objectIn.readObject();

while(obj!=null)

{

MyLine myline = (MyLine)obj;

g2.setColor(myline.lineColor);

//lineList.add(myline);

g2.draw(myline.line);

obj = objectIn.readObject();

}

g2.setColor(color);

objectIn.close();

} catch (Exception e2) {

// TODO: handle exception

}

});

//文件保存

JMenuItem item5=new JMenuItem("save");

item5.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK\_P,ActionEvent.CTRL\_MASK));

item5.addActionListener(e->{

try {

FileOutputStream fileOut = new FileOutputStream(file);

ObjectOutputStream objectOut = new ObjectOutputStream(fileOut);

for(MyLine myline:lineList)

{

objectOut.writeObject(myline);

}

objectOut.close();

}

catch (Exception exception) {

System.out.println(exception.getMessage());

}

});

menubar.add(menu1);

menubar.add(menu2);

menu2.add(item1);

menu2.add(item2);

menu2.add(item3);

menu1.add(item4);

menu1.add(item5);

setJMenuBar(menubar);

//尺寸

setSize(500,500);

setLocation(100,100);

this.add(p);

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

setVisible(true);

}

//重写paint方法，画图

public void paint(Graphics g) {

//super.paintComponents(g);

//super.paint(g);

//设置画笔颜色

g.setColor(a);

//画一个实心圆

g.fillOval(x, y, 5, 5);

Graphics2D g2=(Graphics2D)g;

//画笔大小

BasicStroke bs\_1=new BasicStroke(3,BasicStroke.CAP\_BUTT,BasicStroke.JOIN\_BEVEL);

g2.setStroke(bs\_1);

Point2D p0=new Point2D.Double(this.x,this.y);

Point2D p1=new Point2D.Double(this.x1,this.y1);

//g.drawLine(x, y, x1, y1);

Shape shape;

shape=new Line2D.Double(p0,p1);

g2.draw(shape);

myline=new MyLine(a,new Line2D.Double(p0,p1));

lineList.add(myline);

//g2.drawLine(x, y, x1, y1);

this.x=x1;

this.y=y1;

}

public void mouseClicked(MouseEvent e) {

}

//鼠标点击画图

public void mousePressed(MouseEvent e) {

if(e.getClickCount()==2)

{

x=0;

te=false;

}

else

{

te=true;

}

if(te)

{

if(x!=0)

{

this.x1=e.getX()+10;

this.y1=e.getY()+50;

System.out.println(x+","+y);

this.repaint();

}

else

{

this.x=e.getX()+10;

this.y=e.getY()+50;

}

}

}

public void mouseReleased(MouseEvent e) {

}

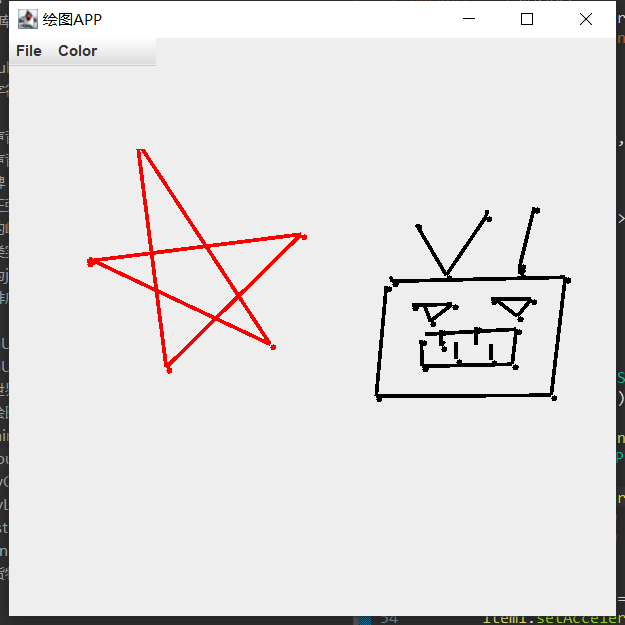
public void mouseEntered(MouseEvent e) {

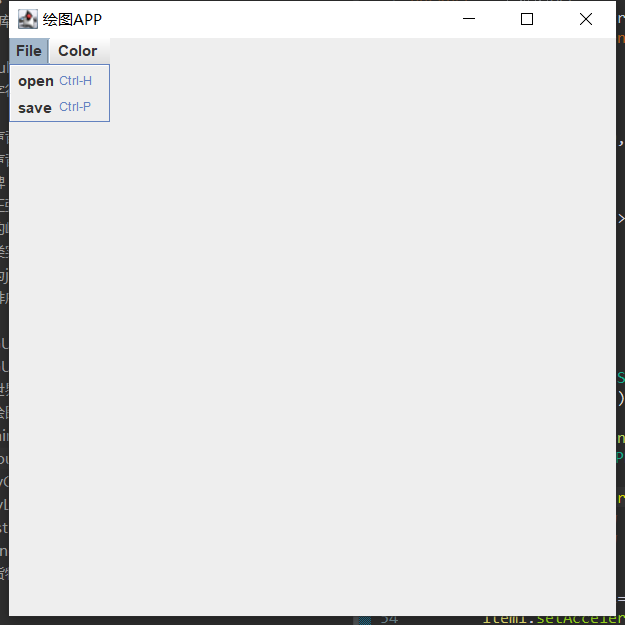
}

public void mouseExited(MouseEvent e) {

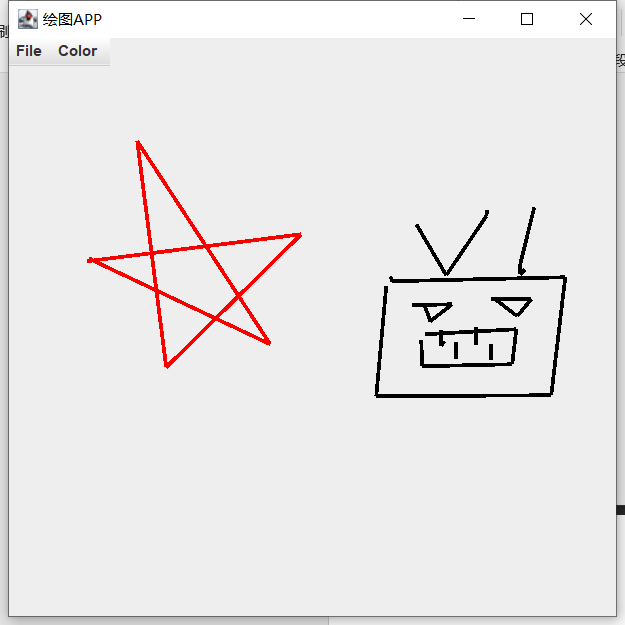
}

}





打开：



加行数：

package 批量为java源码加行号;

import java.io.\*;

public class adddir {

public String dir="D:\\应用\\eclipse\\eclipse-workspace\\MyProjects\\src";

public String temp="D:\\应用\\eclipse\\eclipse-workspace\\MyProjects\\";

public String st="D:\\应用\\eclipse\\eclipse-workspace\\MyProjects\\new";

adddir()

{

addLineNum(dir);

}

public void addLineNum(String s)

{

File f=new File(s);

if(f.isDirectory())

{

//在new文件夹下创建相同的目录

String Str=st+f.getAbsolutePath().substring(dir.length());

File ht=new File(Str);

ht.mkdirs();

System.***out***.println(s+": 添加成功");

//读取f中的文件列表，递归调用addLineNum()去处理每一个表项

//遍历所有文件

File [] allf=f.listFiles();

//递归调用

for(File fi:allf)

{

addLineNum(fi.getAbsolutePath());

}

}

else if(f.getName().endsWith(".java"))

{

//按行复制文件，并加行号。

//创建新的文件(java)

String Str=st+f.getAbsolutePath().substring(dir.length());

File fwrite=new File(Str);

try {

//缓冲流

Writer out=new FileWriter(fwrite);

BufferedWriter bufferWrite=new BufferedWriter(out);

Reader in=new FileReader(f);

BufferedReader bufferRead=new BufferedReader(in);

String str=null;

//给源代码加行数,按行读取

int n=1;

while((str=bufferRead.readLine())!=null)

{

str="行数:"+n+" "+str;

n++;

bufferWrite.write(str);

bufferWrite.newLine();

}

//手动关闭

bufferWrite.close();

out.close();

bufferRead.close();

in.close();

System.***out***.println(s+": 添加成功");

}

catch(IOException e)

{

System.***out***.println(e.toString());

}

}

}

}

