/\*\*

\* PaintWindow.java

\*

\* @author rbender, mlucas, lmannion

\* @version 5/17/18

\* @contact bendryan14@qcsdstu.org

\*

\* Code Description: Will initiate applet characteristics, including panels, JRadioButtons

\* and JButtons, ActionListeners to perform actions based on the state of said buttons, as well

\* as a DrawPad of which the user will draw on

\*

\*/

import java.awt.Container;

import java.awt.Dimension;

import javax.swing.ButtonGroup;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JRadioButton;

import java.awt.BorderLayout;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.Color;

public class PaintWindow extends JFrame

{

//ActionEvent object created

ActionEvent event;

//ButtonGroup’s instantiated

ButtonGroup colorGroup = new ButtonGroup();

ButtonGroup sizeGroup = new ButtonGroup();

//JRadioButtons created for bottomPanel

JRadioButton red = new JRadioButton("Red");

JRadioButton orange = new JRadioButton("Orange");

JRadioButton yellow = new JRadioButton("Yellow");

JRadioButton green = new JRadioButton("Green");

JRadioButton blue = new JRadioButton("Blue");

JRadioButton indigo = new JRadioButton("Indigo");

JRadioButton purple = new JRadioButton("Purple");

JRadioButton black = new JRadioButton("Black");

//JRadioButtons created for sidePanel

JRadioButton smallSize = new JRadioButton("Small Size");

JRadioButton mediumSize = new JRadioButton("Medium Size");

JRadioButton largeSize = new JRadioButton("Large Size");

JRadioButton extraLargeSize = new JRadioButton("Extra-Large Size");

//JButton created for clearing drawPad

JButton clear = new JButton("Clear");

//drawPad created

DrawPad drawPad;

//Color objects created

final Color INDIGO = new Color(75, 0, 130);

final Color VIOLET = new Color( 128, 0, 128 );

public PaintWindow()

{

//Creates 2 options panels and instantiates a new DrawPad

JPanel sidePanel = new JPanel();

JPanel bottomPanel = new JPanel();

drawPad = new DrawPad();

//sets drawPad background color

drawPad.setBackground(Color.white);

//Creates a new container

Container content = this.getContentPane();

//sets the layout

content.setLayout(new BorderLayout());

//sets the panel to the left, padDraw in the center

content.add(bottomPanel, BorderLayout.SOUTH);

content.add(sidePanel, BorderLayout.WEST);

content.add(drawPad, BorderLayout.CENTER);

//size buttons added to size button group

sizeGroup.add(smallSize);

sizeGroup.add(mediumSize);

sizeGroup.add(largeSize);

sizeGroup.add(extraLargeSize);

//action listeners added to size buttons

smallSize.addActionListener(new JButtonListener());

mediumSize.addActionListener(new JButtonListener());

largeSize.addActionListener(new JButtonListener());

extraLargeSize.addActionListener(new JButtonListener());

//action listener added to clear button

clear.addActionListener(new JButtonListener());

clear.setForeground(Color.black);

clear.setOpaque(true);

//action listeners and color added to buttons

red.addActionListener(new JButtonListener());

red.setForeground(Color.red);

red.setOpaque(true);

orange.addActionListener(new JButtonListener());

orange.setForeground(Color.orange);

orange.setOpaque(true);

yellow.addActionListener(new JButtonListener());

yellow.setForeground(Color.yellow);

yellow.setOpaque(true);

green.addActionListener(new JButtonListener());

green.setForeground(Color.green);

green.setOpaque(true);

blue.addActionListener(new JButtonListener());

blue.setForeground(Color.blue);

blue.setOpaque(true);

indigo.addActionListener(new JButtonListener());

indigo.setForeground(INDIGO);

indigo.setOpaque(true);

purple.addActionListener(new JButtonListener());

purple.setForeground(VIOLET);

purple.setOpaque(true);

black.addActionListener(new JButtonListener());

black.setForeground(Color.black);

black.setOpaque(true);

//color buttons added to color button group

colorGroup.add(red);

colorGroup.add(orange);

colorGroup.add(yellow);

colorGroup.add(green);

colorGroup.add(blue);

colorGroup.add(indigo);

colorGroup.add(purple);

colorGroup.add(black);

//preferred size set for size buttons

smallSize.setPreferredSize(new Dimension(130,5));

mediumSize.setPreferredSize(new Dimension(130,5));

largeSize.setPreferredSize(new Dimension(130,5));

extraLargeSize.setPreferredSize(new Dimension(130,5));

//preferred size set for clear button

clear.setPreferredSize(new Dimension(130,5));

//buttons are added to sidePanel

sidePanel.setLayout(new GridLayout(15,1));

sidePanel.add(smallSize);

sidePanel.add(mediumSize);

sidePanel.add(largeSize);

sidePanel.add(extraLargeSize);

sidePanel.add(clear);

//buttons are added to bottomPanel

bottomPanel.add(red);

bottomPanel.add(orange);

bottomPanel.add(yellow);

bottomPanel.add(green);

bottomPanel.add(blue);

bottomPanel.add(indigo);

bottomPanel.add(purple);

bottomPanel.add(black);

}

/\*\*

\* Represents the listener for the JRadioButtons and one JButton.

\*

\* @author rbender, mlucas, lmannion

\* @contact bendryan14@qcsdstu.org

\*/

private class JButtonListener implements ActionListener

{

/\*\*

\* Updates the background color of the panel based on

\* which radio button is selected.

\* @ param event

\*/

public void actionPerformed (ActionEvent event)

{

//conditionals used to determine which button has been clicked, to change

//the color, size, or to clear the canvas

if(red.isSelected())

drawPad.setColor(Color.red);

if(orange.isSelected())

drawPad.setColor(Color.orange);

if(yellow.isSelected())

drawPad.setColor(Color.yellow);

if(green.isSelected())

drawPad.setColor(Color.green);

if(blue.isSelected())

drawPad.setColor(Color.blue);

if(indigo.isSelected())

drawPad.setColor(INDIGO);

if(purple.isSelected())

drawPad.setColor(VIOLET);

if(black.isSelected())

drawPad.setColor(Color.black);

if(smallSize.isSelected())

drawPad.setRadius(9);

if(mediumSize.isSelected())

drawPad.setRadius(18);

if(largeSize.isSelected())

drawPad.setRadius(25);

if(extraLargeSize.isSelected())

drawPad.setRadius(45);

if(clear.isSelected())

drawPad.clear();

}

}

}