

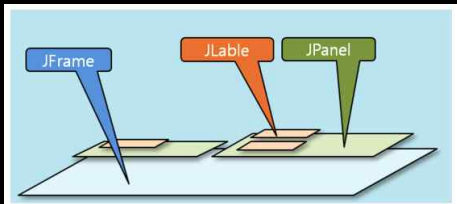
Java로 그림그리기

92313714 함근희

목차

1. JPanel?
2. 그림판 기능
3. 그림판 코드

JPanel이란?



“조각난 프레임”

프레임에서 레이아웃 적용과 각종 UI 배치들을 JPanel에서 다 할 수 있기 때문이다.

1. JFrame : 메인 프레임
2. JPanel : 보조 프레임 (패널이라고도 한다)
3. JLabel 등 각종 UI : 기능을 수행하는 컴포넌트

그림판에 필요한 기능

펜, 지우개, 전체 지우기, 색상 등

```

import java.awt.*;

import java.awt.event.*;

import javax.swing.*;

public class Project extends JFrame {

    JPanel panel, paint;

    JButton pencil, eraser, erase;

    JButton color;

    JLabel thick_label;

    JTextField thick_field; //도구 굵기가 정해질 텍스트 필드

    Color selectedColor;

    Graphics graphics; //Graphics 2D 클래스의 사용을 위해 선언

    Graphics2D g; // 기존 Graphics의 상위버전

```

```

int thick = 10;

int a ;

int startX;

int startY;

int endX;

int endY;

boolean tf = false;

public Project() {

    setLayout(null);

    setTitle("그림판");

    setSize(900,750);

    setLocationRelativeTo(null); //프로그램 시작시 화면 중앙에 출력됨.

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    panel = new JPanel();

    panel.setBackground(Color.GRAY);

    panel.setLayout(null); //panel의 레이아웃을 null로 지정하여 컴포넌트 위치를 직접 지정할 수 있음.

    ImageIcon icon = new ImageIcon("D:\\p.png"); //연필 버튼 생성
    Image img = icon.getImage();

    Image changeimage = img.getScaledInstance(50, 50, Image.SCALE_SMOOTH);

    ImageIcon changeicon = new ImageIcon(changeimage);

    pencil = new JButton(changeicon);

    pencil.setFont(new Font("Serif", Font.BOLD, 25)); //버튼폰트 크기 및 글씨 크기 지정
    pencil.setBackground(Color.LIGHT_GRAY);

```

```

ImageIcon icon2 = new ImageIcon("D:\\e.jpg"); //지우개 버튼 생성
Image img2 = icon2.getImage();
Image changeImage2 = img2.getScaledInstance(50, 50, Image.SCALE_SMOOTH);
ImageIcon changeIcon2 = new ImageIcon(changeImage2);
eraser = new JButton(changeIcon2);

eraser.setFont(new Font("Serif", Font.BOLD, 25)); //버튼 폰트 크기 및 글씨 크기 지정
eraser.setBackground(Color.WHITE);

color = new JButton("색상"); //선 색상 버튼 생성
color.setBackground(Color.YELLOW);

erase = new JButton("모두지우기"); //모두지우기 버튼 생성
erase.setFont(new Font("Serif", Font.BOLD, 25));
erase.setBackground(Color.WHITE);

thick_label = new JLabel("굵기"); //도구 굵기 라벨 지정
thick_label.setFont(new Font("Serif", Font.BOLD, 20)); //도구 굵기 라벨 폰트 및 글씨 크기 지정

thick_field = new JTextField(Integer.toString(thick), 5); // 도구 굵기 입력 텍스트 필드 생성
thick_field.setHorizontalAlignment(JTextField.CENTER);
thick_field.setFont(new Font("Serif", Font.PLAIN, 25));

```

```

pencil.setBounds(10,10,90,55); //연필 버튼 위치 조정
eraser.setBounds(105,10,109,55); //지우개 버튼 위치 조정
erase.setBounds(220,10,180,55); //모두 지우기 버튼 위치 조정
color.setBounds(785,10,90,55); // 선택상 버튼 위치 조정
thick_label.setBounds(640,10,100,55); //도구 굵기 라벨 위치 조정
thick_field.setBounds(720,22,50,35); //도구 굵기 텍스트 필드 위치 조정

```

```

panel.add(pencil); //panel에 연필 버튼 추가
panel.add(eraser); //panel에 지우개 버튼 추가
panel.add(erase); //panel에 모두 지우기 버튼 추가
panel.add(color); //panel에 선택상 버튼 추가
panel.add(thick_label); //panel에 도구굵기 라벨 추가
panel.add(thick_field); //panel에 도구굵기 텍스트 필드 추가
panel.setBounds(0,0,900,75); //panel이 프레임에 배치될 위치 지정

```

```

paint = new JPanel(); //그림이 그려질 패널 생성
paint.setBackground(Color.WHITE);

paint.setLayout(null); //paint의 레이아웃을 null해줘 패널 자체를 setBounds로 위치를 조정 할 수 있도록 함.

paint.setBounds(0,90,885,620); //paint 패널의 위치 조정

```

```

add(panel); //메인프레임에 연필 패널 추가 - 위치는 위에서 다 정해줌.
add(paint); // 메인프레임에 도화지 패널 추가 - 위치는 위에서 다 정해줌

setFocusable(true); //컨텐츠 판이 포커스를 받을 수 있도록 설정
requestFocus(); //컨텐츠판에 포커스 설정
setVisible(true); //메인프레임을 보이게 함.

```

```

graphics = getGraphics(); //그래픽 초기화

g = (Graphics2D)graphics;

g.setColor(selectedColor);

addKeyListener(new KeyListener() {           //key리스너 추가

    public void keyPressed(KeyEvent e) {

        switch(e.getKeyChar()) {

case '[':                //'[' 버튼 누를 시 선 굵기가 10만큼 줄어듦

    a = a-10;

    thick_field.setText(Integer.toString(a));

    break;

case ']':                //']' 버튼 누를 시 선 굵기가 10만큼 늘어남

    a = a+10;

    thick_field.setText(Integer.toString(a));

    break;

case 'q':                //'q'버튼을 누를 시 패널 종료

    System.exit(0);

        }

    }
}

```

```

@Override

public void keyReleased(KeyEvent e) {

    // TODO Auto-generated method stub

}

@Override

public void keyTyped(KeyEvent e) {

    // TODO Auto-generated method stub

}

});

paint.addMouseListener(new MouseListener() { //MouseListener 이벤트 처리

    public void mousePressed(MouseEvent e) {

        startX = e.getX(); //마우스가 눌렀을때 그때의 x좌표값으로 초기화

        startY = e.getY(); //마우스가 눌렀을때 그때의 y좌표값으로 초기화

    }

    public void mouseClicked(MouseEvent e) {} //클릭이벤트 처리

    public void mouseEntered(MouseEvent e) {} //paint(도화지) 범위 내에 진입시 이벤트 처리

    public void mouseExited(MouseEvent e) {}

    public void mouseReleased(MouseEvent e) {}

});

```

```

paint.addMouseMotionListener(new PaintDraw()); //그림 그려질 패널에 마우스 모션 리스너 추가

pencil.addActionListener(new ToolActionListener()); //연필버튼 액션처리

eraser.addActionListener(new ToolActionListener()); //지우개 버튼 이벤트 처리

erase.addActionListener(new ToolActionListener()); //모두지우기 버튼 이벤트 처리


color.addActionListener(new ActionListener() { /// 선택상버튼 액션처리를 익명 클래스로 저장

    public void actionPerformed(ActionEvent e) { //// 오버라이딩

        tf = true;

        JColorChooser chooser = new JColorChooser();

        selectedColor = chooser.showDialog(null, "Color", Color.ORANGE);

        g.setColor(selectedColor);

    }

}); ///

}

public class PaintDraw implements MouseMotionListener{

    public void mouseDragged(MouseEvent e) { //paint 패널에서 마우스 드래그 액션이 처리될때 매소드 실행

        thick = Integer.parseInt(thick_field.getText());

        endX = e.getX();

        endY = e.getY();

```

```

        g.setStroke(new BasicStroke(thick, BasicStroke.CAP_ROUND,0));

        g.drawLine(startX+10, startY+121, endX+10, endY+121);

        startX = endX;

        startY = endY;

    }

    @Override

    public void mouseMoved(MouseEvent e) {}

}

public class ToolActionListener implements ActionListener { // 연필,지우개, 모두지우기 버튼의 액션처리시 실행되는 클래스

    public void actionPerformed(ActionEvent e ) { //오버 라이딩 된 actionPerformed 매소드 실행

        if(e.getSource() == pencil) {

            if(tf == false) g.setColor(Color.BLACK); //우선 색은 검은색으로 지정

            else g.setColor(selectedColor);

        }

        else if(e.getSource() == eraser) {

            g.setColor(Color.WHITE); //그러지는지우개 색상 흰색으로 지정 -> 흰색으로 보이는 것처럼 보이게 함.

        }

        else if(e.getSource() == erase) {

            repaint();

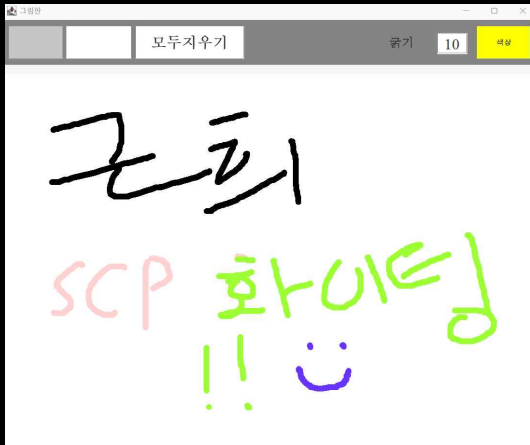
        }

    }

}

```


실행결과



감사합니다