

```
import java.awt.Color;
import java.awt.Graphics;
import java.awt.Point;
public class Line
{
    private Point from;
    private Point to;
    private Color color;

    public Line(int x1, int y1, int x2, int y2)
    {
        from = new Point(x1, y1);
        to = new Point(x2, y2);
    }

    public Line(Point from, Point to, Color color)
    {
        this.from = from;
        this.to = to;
        this.color = color;
    }

    public Point getFrom()
    {
        return from;
    }

    public Point getTo()
    {
        return to;
    }

    public void setFrom(Point from)
    {
        this.from = from;
    }

    public void setTo(Point to)
    {
        this.to = to;
    }

    public void setColor(Color color)
    {
        this.color = color;
    }

    public void draw(Graphics g)
    {
        // Dessiner la ligne dans la couleur de la ligne
        g.setColor(color);
        g.drawLine(from.x, from.y, to.x, to.y);
    }

    public String toString()
    {
        return "[" + from.x + "," + from.y + "] -> [" + to.x + "," + to.y + "];"
    }
}
```

```
import java.awt.Color;
import java.awt.Graphics;
import java.util.ArrayList;
public class Lines
{
    private ArrayList<Line> allLines = new ArrayList<>();

    public void add(Line l)
    {
        allLines.add(l);
    }

    public void clear()
    {
        allLines.clear();
    }

    public void changeLineColor(int i, Color pColor)
    {
        // Ne peut changer la couleur que si la ligne existe
        if (i < allLines.size())
            allLines.get(i).setColor(pColor);
    }

    public void draw(Graphics g)
    {
        // Chaque ligne se dessine elle-même
        for (int i = 0; i < allLines.size(); i++)
            allLines.get(i).draw(g);
    }

    public Object[] toArray()
    {
        return allLines.toArray();
    }
}
```

```
import java.awt.Color;
import java.awt.Graphics;
public class DrawPanel extends javax.swing.JPanel
{
    private Lines lines = null;

    public DrawPanel()
    {
        initComponents();
    }

    public void setLines(Lines lines)
    {
        this.lines = lines;
    }

    @Override
    public void paintComponent(Graphics g)
    {
        // Effacer le dessin
        int w = getWidth();
        int h = getHeight();

        g.setColor(Color.WHITE);
        g.fillRect(0, 0, w, h);

        // Dessiner les lignes si sont présent
        if (lines != null)
            lines.draw(g);
    }
}
// Skipped: ... initComponents { ... }
// Variables declaration - do not modify//GEN-BEGIN:variables
// End of variables declaration//GEN-END:variables
}
```

```

import java.awt.Color;
import javax.swing.JColorChooser;
public class MainFrame extends javax.swing.JFrame
{
    private Lines lines = new Lines();
    private Line newLine = null;

    // Couleur à utiliser pour de nouvelles lignes
    private Color drawColor = Color.BLUE;

    public MainFrame()
    {
        initComponents();
        drawPanel.setLines(lines);
        updateView();
    }

    public void updateView()
    {
        linesList.setListData(lines.toArray());
        colorPanel.setBackground(drawColor);
        repaint();
    }
// Skipped: ... initComponents { ... }
    private void drawPanelMouseDragged(java.awt.event.MouseEvent evt) {GEN-FIRST:event_drawPanelMouseDragged
        newLine.setTo(evt.getPoint());
        updateView();
    }GEN-LAST:event_drawPanelMouseDragged

    private void drawPanelMousePressed(java.awt.event.MouseEvent evt) {GEN-FIRST:event_drawPanelMousePressed
        newLine = new Line(evt.getPoint(), evt.getPoint(), drawColor);
        lines.add(newLine);
        updateView();
    }GEN-LAST:event_drawPanelMousePressed

    private void drawPanelMouseReleased(java.awt.event.MouseEvent evt) {GEN-FIRST:event_drawPanelMouseReleased
        newLine.setTo(evt.getPoint());
        updateView();
    }GEN-LAST:event_drawPanelMouseReleased

    private void changeColorButtonActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_changeColorButtonActionPerformed
    {GEN-HEADEREND:event_changeColorButtonActionPerformed
        drawColor = JColorChooser.showDialog(this, "Choix d'une couleur", drawColor);
        updateView();
    }GEN-LAST:event_changeColorButtonActionPerformed

    private void clearButtonActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_clearButtonActionPerformed
    {GEN-HEADEREND:event_clearButtonActionPerformed
        lines.clear();
        updateView();
    }GEN-LAST:event_clearButtonActionPerformed

    private void changeLineColorButtonActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_changeLineColorButtonActionPerformed
    {GEN-HEADEREND:event_changeLineColorButtonActionPerformed
        Color newColor;
        int i = linesList.getSelectedIndex();
        if (i >= 0)
        {
            newColor = JColorChooser.showDialog(this, "Choix d'une couleur", drawColor);
            lines.changeLineColor(i, newColor);
        }
        updateView();
    }GEN-LAST:event_changeLineColorButtonActionPerformed
// Skipped: ... Look & Feel
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton changeColorButton;
    private javax.swing.JButton changeLineColorButton;
    private javax.swing.JButton clearButton;
    private javax.swing.JPanel colorPanel;
    private DrawPanel drawPanel;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JList linesList;
    // End of variables declaration//GEN-END:variables
}

```