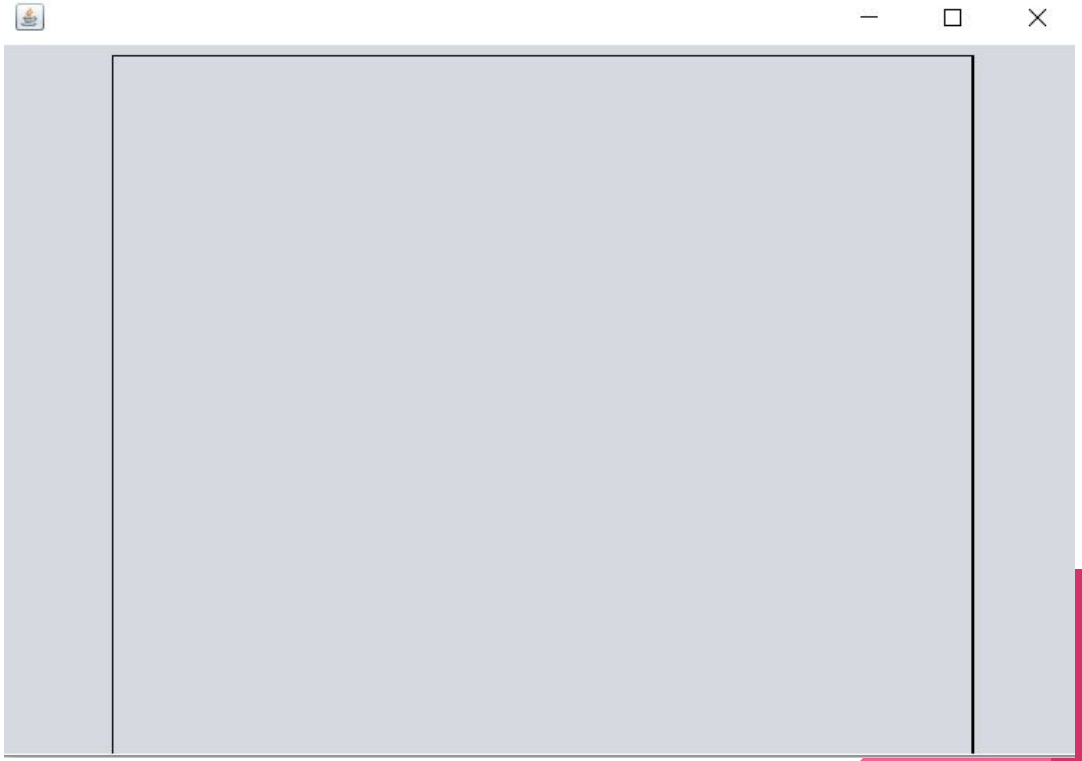
UT5. Edición de gráficos y herramientas de dibujo: BufferedImage y Graphics. Parte 2. Pintar Componentes

José Jarones

JLabel dentro de JPanel



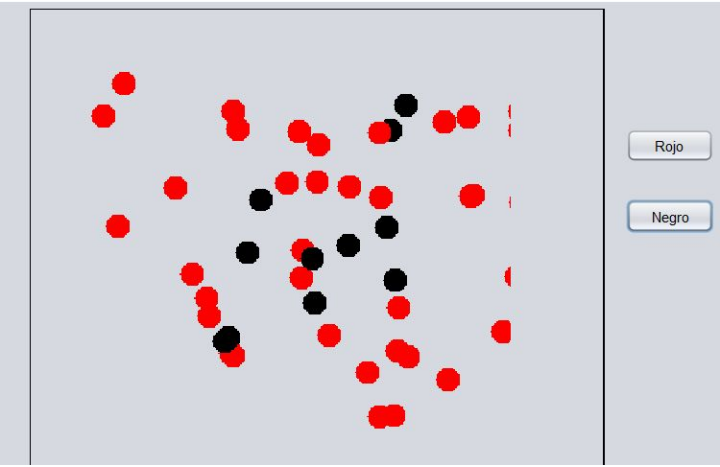
Inicializo componente BufferedImage

```
lic ventana() {
 initComponents();

 BufferedImage buffNuevo = new BufferedImage(400,400,BufferedImage.TYPE_INT_ARGB);
 graficos= buffNuevo.getGraphics();
 this.jLabel1.setIcon(new ImageIcon(buffNuevo));
```

Al presionar botón pinto

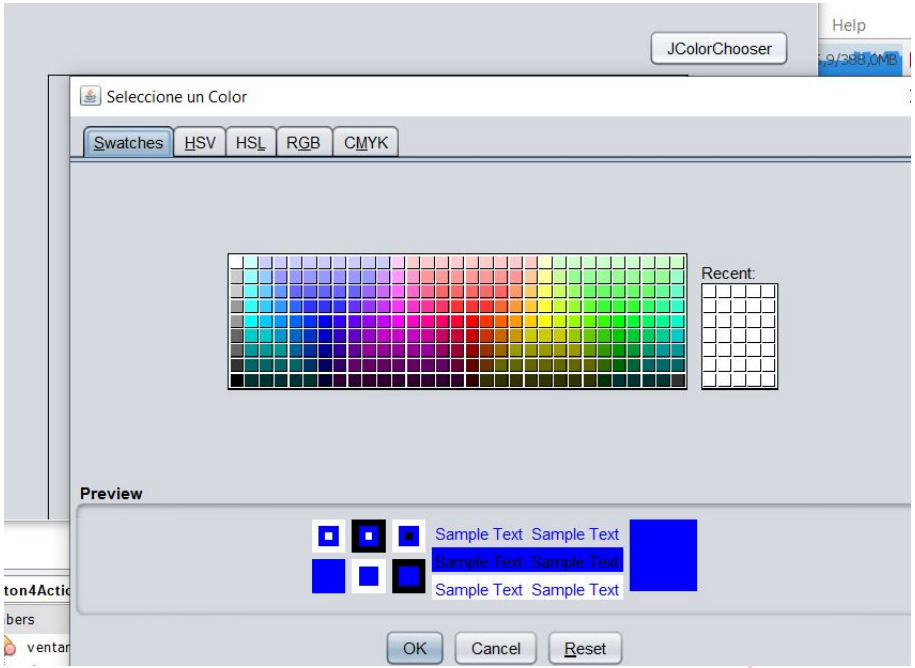
```
private void jPanel1MouseDragged(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
        int x=evt.getX();
    int y= evt.getY();
    graficos.setColor(color);
    graficos.fillOval(x, y, 20, 20);
    System.out.println("Pintando en: x: "+x+" y: "+y);
    this.jLabel1.updateUI();
}
```

Añadir botones con colores

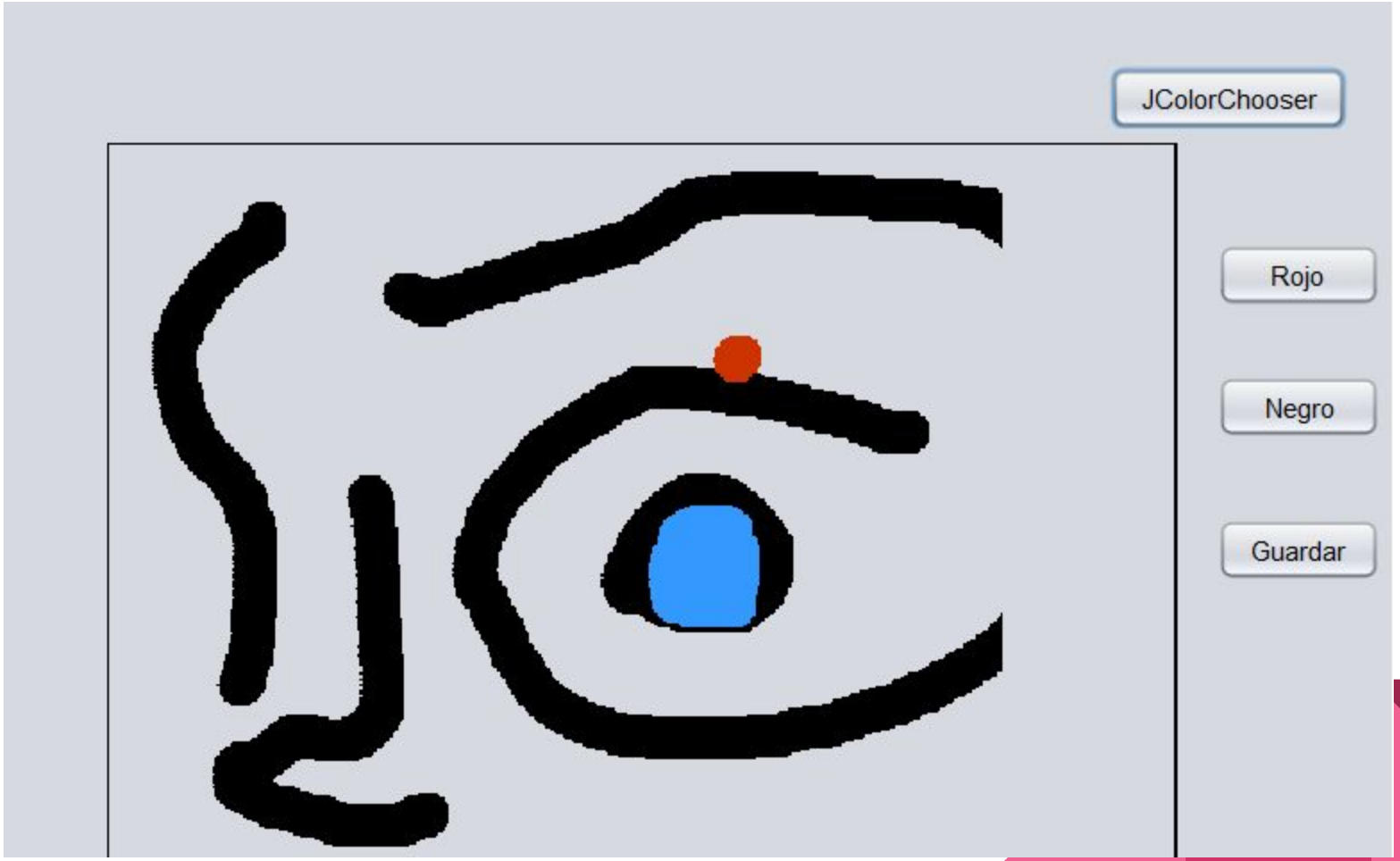



```
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
color=Color.red;         // TODO add your handling code here:
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
color=Color.BLACK;         // TODO add your handling code here:
    }
```

JColorChooser




```
    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
 JColorChooser Selectorcolor=new JColorChooser();
color=Selectorcolor.showDialog(null, "Seleccione un Color", Color.BLUE);
    }
```


JColorChooser
Rojo
Negro
Guardar

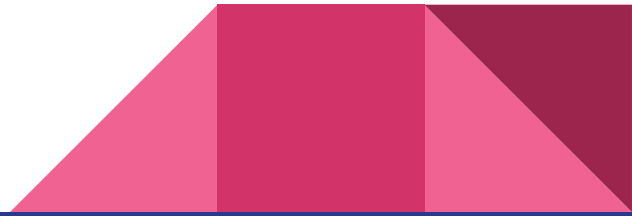
Guardar imagen

```
    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        try {
   ImageIO.write(buffNuevo, "png", new File("foto.png"));
   System.out.println("Imagen guardada");
} catch (IOException e) {
   System.out.println("Error de escritura");
}

    }
```

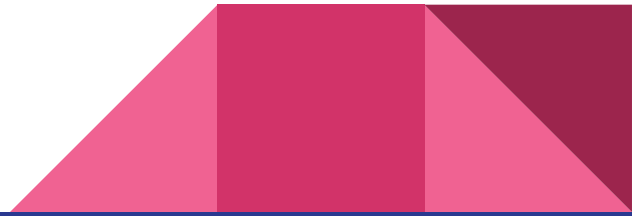
Dibujar

```
//Para poder modificar más propiedades con Graphics 2d
        Graphics2D g2d = (Graphics2D) g;
//Línea
        g2d.setColor(Color.BLUE);
        g2d.setStroke(new BasicStroke(5));
        g2d.drawLine(30, 70, 770, 70);
```

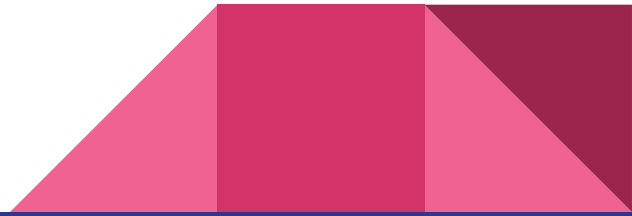
```
//Rectángulo (relleno y borde)
        g2d.setColor(Color.BLUE);
        g2d.fillRect(30, 100, 350, 60);
        g2d.setColor(Color.BLACK);
        g2d.drawRect(30, 100, 350, 60);
```



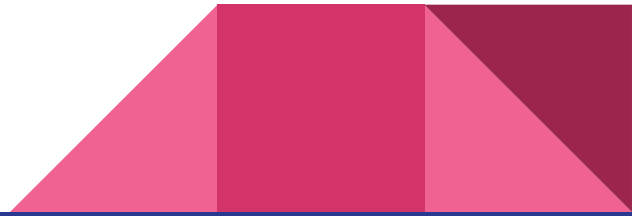
```
//Rectángulo redondeado
        g2d.setColor(Color.CYAN);
        g2d.drawRoundRect (420, 100, 350, 60, 50, 50);

        //Arco
        g2d.setColor(Color.PINK);
        g2d.drawArc(30, 200, 100, 100, 180, -90);
```



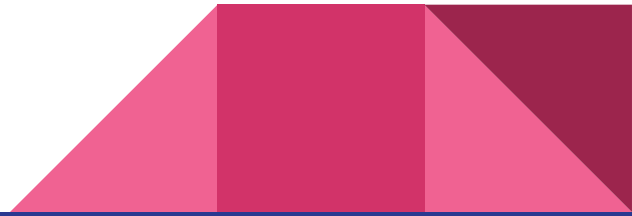
```
//Círculo
        g2d.setColor(Color.RED);
        g2d.drawOval(100, 200, 100, 100);

        //Óvalo (con relleno y borde)
        g2d.setColor(Color.YELLOW);
        g2d.fillOval(240, 200, 150, 100);
        g2d.setColor(Color.BLACK);
        g2d.drawOval(240, 200, 150, 100);
```



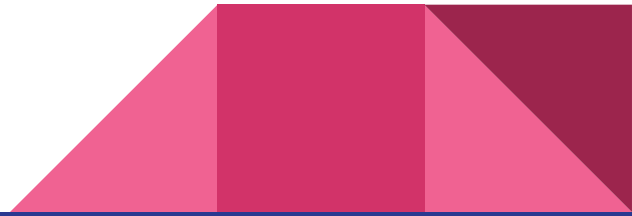
```
//Polígono (3 lados)
    int [] triangulo_x = {450, 510, 570};
    int [] triangulo_y = {300, 200, 300};
    g2d.setColor(Color.ORANGE);
    g2d.drawPolygon (triangulo_x, triangulo_y, 3);
```



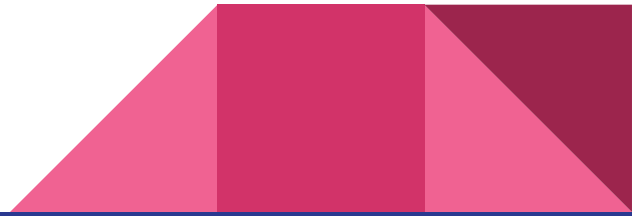
```
//Polígono (5 lados con relleno y borde)
    int [] pentagono_x = {670, 650, 700, 750, 730};
    int [] pentagono_y = {300, 245, 200, 245, 300};
    g2d.setColor(Color.MAGENTA);
    g2d.fillPolygon (pentagono_x, pentagono_y, 5);
    g2d.setColor(Color.BLACK);
    g2d.drawPolygon (pentagono_x, pentagono_y, 5);
```



```
//Texto
        g2d.setColor(Color.BLACK);
        g2d.setFont(new Font("ARIAL",PLAIN,32));
        g2d.drawString("Esto es un texto", 30, 400);
```



```
//Imagen
        Toolkit t = Toolkit.getDefaultToolkit();
        Image imagen = t.getImage ("src/img/smile.png");
        g2d.drawImage(imagen, 30, 450, this);
```



```
//Degradado

        GradientPaint gp = new GradientPaint(400, 350, Color.RED, 770, 550,
Color.GREEN);

        g2d.setPaint(gp);

        g2d.fillRect(400, 350, 370, 200);
```