



UNIVERSIDAD NACIONAL DE COLOMBIA SEDE  
MEDELLÍN

Tarea de programación orientada a objetos – 1  
Octubre 2022

Alumno:

Angello Gómez Monsalve

CC: 1.000.566.629

Profesor:

Walter Hugo Arboleda Mazo

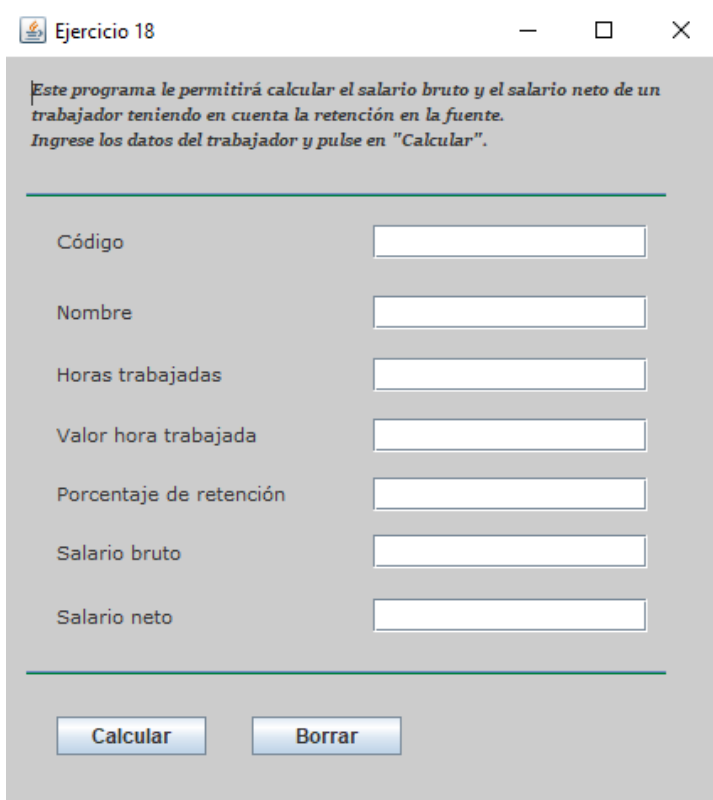
REPOSITORIO EN GITHUB

:<https://github.com/anggomezmo/Tarea-1-octubre>

REPOSITORIO EN GITHUB

## CAPITULO 3

### EJERCICIO PROPUESTO #18:



The screenshot shows a Java Swing window titled "Ejercicio 18". Inside the window, there is a text area with the following text: "Este programa le permitirá calcular el salario bruto y el salario neto de un trabajador teniendo en cuenta la retención en la fuente. Ingrese los datos del trabajador y pulse en 'Calcular'." Below the text area, there are seven text input fields arranged vertically, each with a label to its left: "Código", "Nombre", "Horas trabajadas", "Valor hora trabajada", "Porcentaje de retención", "Salario bruto", and "Salario neto". At the bottom of the window, there are two buttons: "Calcular" and "Borrar".

## Programa main:

```
package Cap3_Ejercicio18;

public class Ejercicio18 {
    public static void main(String[] args) {
//Programa Main
        Formulario_18 fr= new Formulario_18();
        fr.setVisible(true);

    }
}
```

## Formulario:

```
}

package Cap3_Ejercicio18;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
```

```
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JEditorPane;
import javax.swing.JSeparator;
import java.awt.Color;
import java.awt.Font;

public class Formulario_18 extends JFrame {

    private JPanel contentPane;
    private JTextField textCodigo;
    private JTextField textNombre;
    private JTextField textHoras;
    private JTextField textValor;
    private JTextField textRetencion;
    private JTextField textSalariob;
    private JTextField textSalarion;

    /**
     * Launch the application.
     */
}
```

```

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Formulario_18 frame = new
Formulario_18();

                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Formulario_18() {
        this.setTitle("Ejercicio 18");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 440, 485);
        contentPane = new JPanel();
        contentPane.setBackground(new Color(204, 204,
204));

        contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));
    }

```

```
setContentPane(contentPane);
contentPane.setLayout(null);

JLabel lblNewLabel = new JLabel("Código");
lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel.setBounds(30, 102, 89, 14);
contentPane.add(lblNewLabel);

JLabel lblNewLabel_1 = new JLabel("Nombre");
lblNewLabel_1.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel_1.setBounds(30, 144, 89, 14);
contentPane.add(lblNewLabel_1);

JLabel lblNewLabel_2 = new JLabel("Horas
trabajadas");
lblNewLabel_2.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel_2.setBounds(30, 181, 137, 14);
contentPane.add(lblNewLabel_2);

JLabel lblNewLabel_3 = new JLabel("Valor hora
trabajada");
lblNewLabel_3.setFont(new Font("Verdana",
Font.PLAIN, 11));
```

```
lblNewLabel_3.setBounds(30, 217, 137, 14);  
contentPane.add(lblNewLabel_3);
```

```
JLabel lblNewLabel_4 = new JLabel("Porcentaje de  
retención");
```

```
lblNewLabel_4.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_4.setBounds(30, 252, 205, 14);  
contentPane.add(lblNewLabel_4);
```

```
JLabel lblNewLabel_5 = new JLabel("Salario  
bruto");
```

```
lblNewLabel_5.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_5.setBounds(30, 287, 156, 14);  
contentPane.add(lblNewLabel_5);
```

```
JLabel lblNewLabel_6 = new JLabel("Salario  
neto");
```

```
lblNewLabel_6.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_6.setBounds(30, 325, 89, 14);  
contentPane.add(lblNewLabel_6);
```

```
textCodigo = new JTextField();
```

```
textCodigo.setBounds(218, 100, 163, 20);  
contentPane.add(textCodigo);
```

```
textCodigo.setColumns(10);
```

```
textNombre = new JTextField();
```

```
textNombre.setColumns(10);
```

```
textNombre.setBounds(218, 142, 163, 20);
```

```
contentPane.add(textNombre);
```

```
textHoras = new JTextField();
```

```
textHoras.setColumns(10);
```

```
textHoras.setBounds(218, 179, 163, 20);
```

```
contentPane.add(textHoras);
```

```
textValor = new JTextField();
```

```
textValor.setColumns(10);
```

```
textValor.setBounds(218, 215, 163, 20);
```

```
contentPane.add(textValor);
```

```
textRetencion = new JTextField();
```

```
textRetencion.setColumns(10);
```

```
textRetencion.setBounds(218, 250, 163, 20);
```

```
contentPane.add(textRetencion);
```

```
textSalariob = new JTextField();
```

```
textSalariob.setColumns(10);
```

```
textSalariob.setBounds(218, 284, 163, 20);
```



```

        contentPane.add(textSalariob);

        textSalarion = new JTextField();
        textSalarion.setColumns(10);
        textSalarion.setBounds(218, 322, 163, 20);
        contentPane.add(textSalarion);

        JButton btnCalcular = new JButton("Calcular");
        btnCalcular.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
            {
                //CALCULA EL SALARIO BRUTO, NETO, Y
                DINERO RETENIDO

                try {
                    double horas, valorhora,
                    porcentaje_retencion, dinero_retenido;

                    horas=Double.parseDouble(textHoras.getText());

                    valorhora=Double.parseDouble(textValor.getText());

                    porcentaje_retencion=Double.parseDouble(textRetencio
                    n.getText());

```

```
        textSalariob.setText(String.valueOf(Métodos.metodo_salario_bruto(horas, valorhora)));
```

```
        dinero_retenido=Métodos.metodo_retencion_fuente(horas, valorhora, porcentaje_retencion);
```

```
        textSalarion.setText(String.valueOf(Métodos.metodo_salario_netto(dinero_retenido, horas, valorhora)));
```

```
    }
```

```
        catch(NumberFormatException ex) {
```

```
            //EXCEPCION VALORES NO NÚMERICOS
```

```
        JOptionPane.showMessageDialog(null, "Rellene todos los campos o utilice un formato valido");
```

```
    }
```

```
}
```

```
});
```

```
btnCalcular.setBounds(30, 392, 89, 23);
```

```
contentPane.add(btnCalcular);
```

```
JButton btnBorrar = new JButton("Borrar");
```

```
btnBorrar.addActionListener(new ActionListener()
```

```
{
```

```

        public void actionPerformed(ActionEvent e)
    {
        //Limpia todos los campos Jfield

        textSalariob.setText("");
        textSalarion.setText("");
        textCodigo.setText("");
        textNombre.setText("");
        textHoras.setText("");
        textRetencion.setText("");
        textValor.setText("");
    }
});
btnBorrar.setBounds(146, 392, 89, 23);
contentPane.add(btnBorrar);

JEditorPane dtrpnEsteProgramaLe = new
JEditorPane();
dtrpnEsteProgramaLe.setBackground(new Color(204,
204, 204));
dtrpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 11));
dtrpnEsteProgramaLe.setText("Este programa le
permitirá calcular el salario bruto y el salario neto de
un \r\ntrabajador teniendo en cuenta la retención en la
fuente.\r\nIngrese los datos del trabajador y pulse en
\"Calcular\".");
dtrpnEsteProgramaLe.setBounds(12, 11, 402, 59);

```

```

        contentPane.add(dtrpnEsteProgramaLe);

        JSeparator separator = new JSeparator();
        separator.setBackground(new Color(0, 128, 64));
        separator.setBounds(12, 365, 380, 2);
        contentPane.add(separator);

        JSeparator separator_1 = new JSeparator();
        separator_1.setBackground(new Color(0, 128,
64));
        separator_1.setBounds(12, 68, 380, 2);
        contentPane.add(separator_1);

        JSeparator separator_2 = new JSeparator();
        separator_2.setBackground(new Color(0, 128,
64));
        separator_2.setBounds(12, 81, 380, 10);
        contentPane.add(separator_2);
    }
}

```

## Métodos:

```
package Cap3_Ejercicio18;
```

```
public class Métodos {

    public static double metodo_salario_bruto(double
horas, double valorhora) {

        //Metodo que calcula el salario bruto
        double salario_bruto;
        salario_bruto= horas*valorhora;
        return salario_bruto;

    }

    public static double metodo_retencion_fuente(double
porcentaje_retencion,double horas, double valorhora) {

        //Metodo que calcula el dinero retenido en la
fuente

        double dinero_retenido;

        dinero_retenido=
(horas*valorhora)*(porcentaje_retencion)/100;

        return dinero_retenido;

    }

    public static double metodo_salario_netto(double
dinero_retenido,double horas, double valorhora) {

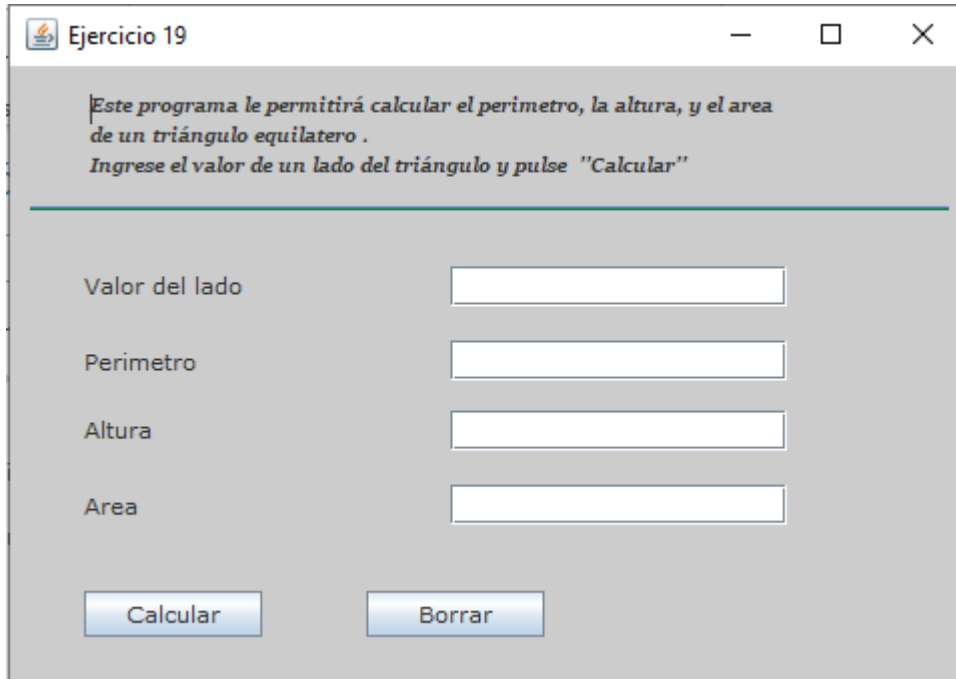
        //metodo que calcula el salario netto
        double salario_netto;
        salario_netto= (horas*valorhora)-
dinero_retenido;

    }

}
```

```
        return salario_netto;  
    }  
  
}
```

## EJERCICIO RESUELTO #19:



**Ejercicio 19**

*Este programa le permitirá calcular el perímetro, la altura, y el área de un triángulo equilátero .  
Ingrese el valor de un lado del triángulo y pulse "Calcular"*

---

Valor del lado	<input type="text"/>
Perímetro	<input type="text"/>
Altura	<input type="text"/>
Área	<input type="text"/>

## Programa main:

```
package Cap3_Ejercicio19;  
  
public class Ejercicio19 {
```

```
        public static void main(String[] args) { //Metodo
main            Formulario_19 fr= new Formulario_19();
                fr.setVisible(true);

        }
    }
```

## Formulario:

```
package Cap3_Ejercicio19;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JOptionPane;
import javax.swing.JEditorPane;
import java.awt.Font;
import javax.swing.JSeparator;
import java.awt.Color;
```

```

public class Formulario_19 extends JFrame {

    private JPanel contentPane;
    private JTextField txtLado;
    private JTextField txtPerimetro;
    private JTextField txtArea;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Formulario_19 frame = new
Formulario_19();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**

```



```

    * Create the frame.
    */
    public Formulario_19() {
        this.setTitle("Ejercicio 19");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 495, 352);
        contentPane = new JPanel();
        contentPane.setBackground(new Color(204, 204,
204));
        contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

        setContentPane(contentPane);
        contentPane.setLayout(null);

        JLabel lblNewLabel = new JLabel("Valor del
lado");
        lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel.setBounds(37, 102, 138, 14);
        contentPane.add(lblNewLabel);

        JLabel lblNewLabel_1 = new JLabel("Perimetro");
        lblNewLabel_1.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_1.setBounds(37, 140, 104, 14);
        contentPane.add(lblNewLabel_1);
    }
}

```

```
JLabel lblNewLabel_2 = new JLabel("Altura");  
lblNewLabel_2.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_2.setBounds(37, 174, 89, 14);  
contentPane.add(lblNewLabel_2);
```

```
JLabel lblNewLabel_3 = new JLabel("Area");  
lblNewLabel_3.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_3.setBounds(37, 212, 46, 14);  
contentPane.add(lblNewLabel_3);
```

```
txtLado = new JTextField();  
txtLado.setBounds(220, 100, 168, 20);  
contentPane.add(txtLado);  
txtLado.setColumns(10);
```

```
txtPerimetro = new JTextField();  
txtPerimetro.setColumns(10);  
txtPerimetro.setBounds(220, 137, 168, 20);  
contentPane.add(txtPerimetro);
```

```
JTextField txtAltura = new JTextField();  
txtAltura.setColumns(10);  
txtAltura.setBounds(220, 172, 168, 20);
```

```

        contentPane.add(txtAltura);

        txtArea = new JTextField();
        txtArea.setColumns(10);
        txtArea.setBounds(220, 209, 168, 20);
        contentPane.add(txtArea);

        JButton btnCalcular = new JButton("Calcular");
        btnCalcular.setFont(new Font("Verdana",
Font.PLAIN, 11));
        btnCalcular.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
        {
            //Programacion del codigo de boton
            Calcular

            double lado;

            try {
                //Calculo de las propiedades del
            triangulo.

            lado =
            Double.parseDouble(txtLado.getText());

            txtPerimetro.setText(String.valueOf(Métodos.calcular
            _perimetro(lado)));

```

```
txtAltura.setText(String.valueOf(Métodos.calcular_al  
tura(lado)));
```

```
txtArea.setText(String.valueOf(Métodos.calcular_area  
(lado)));
```

```
    }catch(NumberFormatException ex) {  
        //EXCEPCION VALORES NO NÚMERICOS
```

```
JOptionPane.showMessageDialog(null, "Rellene el  
campo 'Valor del lado' o utilice un formato valido");
```

```
    }
```

```
    }
```

```
});
```

```
btnCalcular.setBounds(37, 262, 89, 23);
```

```
contentPane.add(btnCalcular);
```

```
JButton btnBorrar = new JButton("Borrar");
```

```
btnBorrar.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
btnBorrar.addActionListener(new ActionListener()  
{  
    public void actionPerformed(ActionEvent e)  
    {
```

```

        //Codigo del boton Borrar, el cual
limpia campos.

        txtPerimetro.setText("");
        txtAltura.setText("");
        txtArea.setText("");
        txtLado.setText("");

    }

});

btnBorrar.setBounds(178, 262, 89, 23);
contentPane.add(btnBorrar);

JEditorPane dtrpnEsteProgramaLe = new
JEditorPane();

dtrpnEsteProgramaLe.setBackground(new Color(204,
204, 204));

dtrpnEsteProgramaLe.setText("Este programa le
permitirá calcular el perimetro, la altura, y el
area\r\nde un triángulo equilatero .\r\nIngrese el valor
de un lado del triángulo y pulse  ''Calcular''");

dtrpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 11));

dtrpnEsteProgramaLe.setBounds(37, 11, 402, 48);
contentPane.add(dtrpnEsteProgramaLe);

JSeparator separator = new JSeparator();

```

```
        separator.setBackground(new Color(0, 128, 64));
        separator.setBounds(10, 70, 459, 21);
        contentPane.add(separator);
    }

}
```

### -Metodos:

```
package Cap3_Ejercicio19;

import java.text.DecimalFormat;

public class Métodos {
    //Metodo que calcula el perimetro del triángulo
    public static double calcular_perimetro(double lado)
    {
        double perimetro;
        perimetro=lado*3;
        return perimetro;

        //Metodo que calcula la altura del triángulo
    }
    public static String calcular_altura(double lado) {
```

```
        String altura;
        DecimalFormat df = new DecimalFormat("#.0000");
        altura=df.format((lado*Math.sqrt(3))/2);
        return altura;
    }
    //metodo que calcula el area del triángulo
    public static double calcular_area(double lado) {
        double area;
        area=(Math.pow(lado,2)*Math.sqrt(3))/4;
        return area;
    }
}
```

## CAPITULO 4

### EJERCICIO #7:

*Este programa le permitirá saber cual de dos numeros es el mayor.*

---

Ingresa valor de A

Ingresa valor de B

**RESULTADO**

---

Programa main:

```
package Cap4_Ejercicio7;
```

```
public class Ejercicio7 {
```



```
//metodo main
public static void main(String[] args) {
    Formulario7 fr= new Formulario7();
    fr.setVisible(true);
}
}
```

## Formulario:

```
package Cap4_Ejercicio7;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;

import javax.swing.JTextField;
import javax.swing.JButton;
import javax.swing.JEditorPane;
import java.awt.Color;
import javax.swing.JSeparator;

public class Formulario7 extends JFrame {

    private JPanel contentPane;
    private JTextField textA;
    private JTextField textB;
    private JTextField textResultado;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Formulario7 frame = new
Formulario7();

                    frame.setVisible(true);
```

```

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
});
}

/**
 * Create the frame.
 */
public Formulario7() {
    this.setTitle("Ejercicio 7");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 502, 359);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(204, 204,
204));
    contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    JLabel lblNewLabel_1 = new JLabel("Ingrese valor
de A");

```

```
        lblNewLabel_1.setFont(new Font("Verdana",
Font.PLAIN, 15));
        lblNewLabel_1.setBounds(40, 96, 170, 14);
        contentPane.add(lblNewLabel_1);

        textA = new JTextField();
        textA.setBounds(201, 95, 86, 20);
        contentPane.add(textA);
        textA.setColumns(10);

        textB = new JTextField();
        textB.setText("");
        textB.setBounds(201, 128, 86, 20);
        contentPane.add(textB);
        textB.setColumns(10);

        JButton btnComparar = new JButton("Comparar");

        btnComparar.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
        {

            try {
```

```

        //Calculo del numero mayor
        double A;
        double B;
        A =
Double.parseDouble(textA.getText());
        B =
Double.parseDouble(textB.getText());

        if
(metodos7.Calcular_mayor_numero(A,B) == A) {
            textResultado.setText("A es
mayor que B");
        }
        else if
(metodos7.Calcular_mayor_numero(A,B) == B) {
            textResultado.setText("B es
mayor que A");
        }
        else {
            textResultado.setText("A es
igual a B");;
        }
    }
    catch(NumberFormatException ex) {
        //EXCEPCION VALORES NO
NÚMERICOS

```

```
JOptionPane.showMessageDialog(null, "Los parámetros  
ingresados no son numéricos");
```

```
}
```

```
}
```

```
});
```

```
btnComparar.setBounds(229, 190, 89, 23);
```

```
contentPane.add(btnComparar);
```

```
textResultado = new JTextField();
```

```
textResultado.setBounds(29, 217, 170, 20);
```

```
contentPane.add(textResultado);
```

```
textResultado.setColumns(10);
```

```
JButton btnBorrar = new JButton("Borrar");
```

```
btnBorrar.addActionListener(new ActionListener()
```

```
{
```

```
    public void actionPerformed(ActionEvent e)
```

```
{
```

```
        //CONFIGURA EL BOTON BORRAR
```

```
        textResultado.setText("");
```

```
        textA.setText("");
```

```

        textB.setText("");
    }
});
btnBorrar.setBounds(229, 224, 89, 23);
contentPane.add(btnBorrar);

JLabel lblNewLabel_10 = new JLabel("RESULTADO");
lblNewLabel_10.setFont(new Font("Arial Rounded
MT Bold", Font.PLAIN, 15));
lblNewLabel_10.setBounds(63, 192, 156, 14);
contentPane.add(lblNewLabel_10);

JEditorPane dtrpnEsteProgramaLe = new
JEditorPane();
dtrpnEsteProgramaLe.setBackground(new Color(204,
204, 204));
dtrpnEsteProgramaLe.setForeground(new Color(0,
0, 0));
dtrpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 14));
dtrpnEsteProgramaLe.setText("Este programa le
permitirá saber cual de dos numeros\r\n es el mayor.");
dtrpnEsteProgramaLe.setBounds(27, 11, 364, 42);
contentPane.add(dtrpnEsteProgramaLe);

JLabel lblNewLabel_1_1 = new JLabel("Ingrese
valor de B\r\n");

```

```

        lblNewLabel_1_1.setFont(new Font("Verdana",
Font.PLAIN, 15));
        lblNewLabel_1_1.setBounds(40, 129, 170, 14);
        contentPane.add(lblNewLabel_1_1);

        JSeparator separator = new JSeparator();
        separator.setBackground(new Color(51, 102, 51));
        separator.setBounds(10, 62, 441, 14);
        contentPane.add(separator);

        JSeparator separator_1 = new JSeparator();
        separator_1.setBackground(new Color(51, 102,
51));
        separator_1.setBounds(10, 267, 441, 14);
        contentPane.add(separator_1);
    }
}

```

## Metodos:

```
package Cap4_Ejercicio7;
```

```
public class metodos7 {
```

```
    //METODO QUE COMPARA DOS NUMEROS Y DECIDE CUAL ES EL
    AMYOR
```

```
    public static double Calcular_mayor_numero(double A,double B) {
```



```
if (A>B) {
```

```
    return A;  //A MAYO QUE B
```

```
}else{
```

```
    if (A<B) {
```

```
        //B MAYOR QUE A
```

```
        return B;
```

```
    }else {
```

```
        return 0;    //SON IGUALES
```

```
    }
```

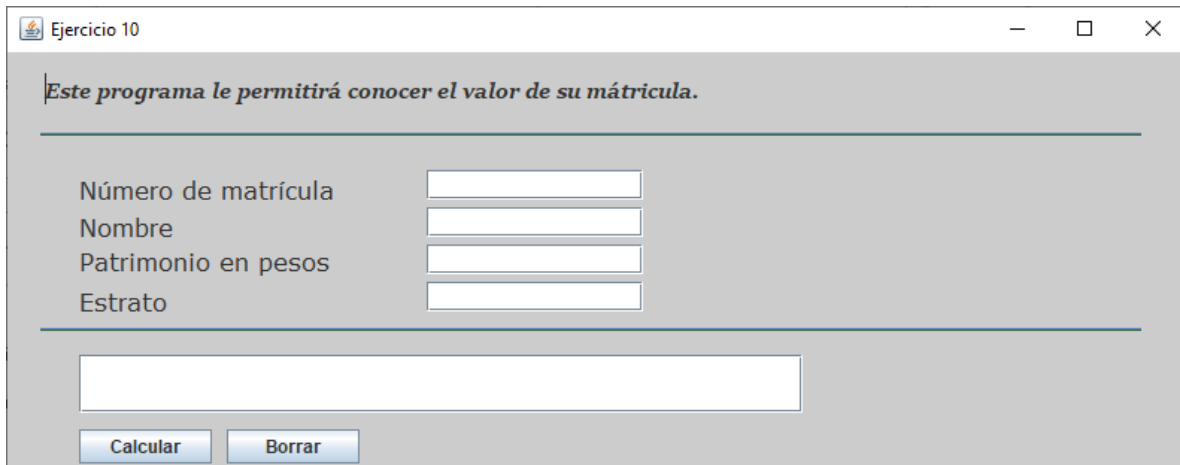
```
}
```

```
}
```

```
}
```

```
}
```

## EJERCICIO #10:



Ejercicio 10

Este programa le permitirá conocer el valor de su matrícula.

Número de matrícula

Nombre

Patrimonio en pesos

Estrato

Calcular Borrar

Programa main:

```
package Cap4_Ejercicio10;
```

```
public class Ejercicio10 {  
  
    //Metodo main  
    public static void main(String[] args) {  
        Formulario10 fr= new Formulario10();  
        fr.setVisible(true);  
    }  
}
```

FORMULARIO:

```

package Cap4_Ejercicio10;

import java.awt.EventQueue;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.JButton;
import javax.swing.JTextPane;
import java.awt.Color;
import javax.swing.JSeparator;
import java.awt.Font;

public class Formulario10 extends JFrame {

    private JPanel contentPane;
    private JTextField textResultado;
    private JTextField textNumero;
    private JTextField textNombre;
    private JTextField textPlata;
    private JTextField textEst;
    private JButton btnBorrar;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {

```

```

        Formulario10 frame = new
Formulario10();
        frame.setVisible(true);
    } catch (Exception e) {
        e.printStackTrace();
    }
    });
}

/**
 * Create the frame.
 */
public Formulario10() {
    this.setTitle("Ejercicio 10");

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 807, 326);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(204, 204,
204));
    contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    JLabel lblNewLabel = new JLabel("Número de
matrícula");
    lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 16));
    lblNewLabel.setBounds(48, 85, 207, 14);
    contentPane.add(lblNewLabel);

    JLabel lblNewLabel_1 = new JLabel("Nombre");
    lblNewLabel_1.setFont(new Font("Verdana",
Font.PLAIN, 16));
    lblNewLabel_1.setBounds(48, 110, 162, 14);
    contentPane.add(lblNewLabel_1);

```

```

        JLabel lblNewLabel_2 = new JLabel("Patrimonio en
pesos");
        lblNewLabel_2.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_2.setBounds(48, 129, 195, 23);
        contentPane.add(lblNewLabel_2);

        JLabel lblNewLabel_3 = new JLabel("Estrato");
        lblNewLabel_3.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_3.setBounds(48, 160, 177, 14);
        contentPane.add(lblNewLabel_3);

        textResultado = new JTextField();
        textResultado.setFont(new Font("Verdana",
Font.PLAIN, 14));
        textResultado.setBounds(48, 203, 485, 39);
        contentPane.add(textResultado);
        textResultado.setColumns(10);

        JButton btnNewButton = new JButton("Calcular");
        btnNewButton.addActionListener(new
ActionListener() {
            public void actionPerformed(ActionEvent e)
            {
                try {
                    //Calculo de Matricula
                    double numero;
                    String nombre;
                    double plata;
                    double estrato;

                    nombre = textNombre.getText();
                    numero =
Double.parseDouble(textNumero.getText());
                    int Hola = (int)numero;
                    estrato =
Double.parseDouble(textEst.getText());

```

```

        int Hola2 = (int)estrato;
        plata =
Double.parseDouble(textPlata.getText());
        textResultado.setText("el
estudiante "+nombre+", con matricula "+Hola+", debe pagar
"+metodos10.Calcular_matricula(plata,estrato));
    }

```

```

        catch(NumberFormatException ex) {
            //EXCEPCION VALORES NO

```

NÚMERICOS

```

        JOptionPane.showMessageDialog(null, "Los parámetros
ingresados no son numéricos");

```

```

    }

```

```

}

```

```

    });

```

```

    btnNewButton.setBounds(48, 253, 89, 23);
    contentPane.add(btnNewButton);

```

```

    textNumero = new JTextField();
    textNumero.setBounds(281, 79, 145, 20);
    contentPane.add(textNumero);
    textNumero.setColumns(10);

```

```

    textNombre = new JTextField();
    textNombre.setBounds(281, 104, 145, 20);
    contentPane.add(textNombre);
    textNombre.setColumns(10);

```

```

    textPlata = new JTextField();
    textPlata.setBounds(281, 129, 145, 20);
    contentPane.add(textPlata);

```

```

textPlata.setColumns(10);

textEst = new JTextField();
textEst.setBounds(281, 154, 145, 20);
contentPane.add(textEst);
textEst.setColumns(10);

btnBorrar = new JButton("Borrar");
btnBorrar.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        //CONFIGURA EL BOTON BORRAR
        textEst.setText("");
        textPlata.setText("");
        textNombre.setText("");
        textNumero.setText("");
        textResultado.setText("");
    }
});
btnBorrar.setBounds(147, 253, 89, 23);
contentPane.add(btnBorrar);

JTextPane txtpnEsteProgramaLe = new JTextPane();
txtpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 16));
txtpnEsteProgramaLe.setBackground(new Color(204,
204, 204));
txtpnEsteProgramaLe.setText("Este programa le
permitirá conocer el valor de su matrícula.");
txtpnEsteProgramaLe.setBounds(22, 11, 537, 39);
contentPane.add(txtpnEsteProgramaLe);

JSeparator separator = new JSeparator();
separator.setBackground(new Color(51, 102, 51));
separator.setBounds(22, 54, 738, 14);
contentPane.add(separator);

JSeparator separator_1 = new JSeparator();

```

```

        separator_1.setBackground(new Color(51, 102,
51));
        separator_1.setBounds(22, 185, 738, 14);
        contentPane.add(separator_1);
    }
}

```

## Métodos:

```
package Cap4_Ejercicio10;
```

```
public class metodos10 {
```

```
    //METODO QUE SE ENCARGA DE CALCULAR EL PAGO DE LA
    MATRICULA.
```

```
    public static double Calcular_matricula(double
plata,double estrato) {
```

```
        double pago;
```

```
        double normal=50000;
```

```
        pago = 0;
```

```
        if ((plata > 2000000) && (estrato > 3)) {
```

```
            pago = 0.03*plata+normal;
```

```
            return pago;
```

```
        }else {
```

```
            pago = normal;
```

```
            return pago;
```



```
}  
}  
}
```

## EJERCICIO #22:

***Este programa le permitirá calcular el salario de un empleado siempre y cuando sea mayor a 450.000***

---

Ingrese nombre

Ingrese salario por hora

Ingrese cantidad horas trabajadas al mes

Programa main:

```
package Cap_Ejercicio22;
```

```
public class Ejercicio22 {  
    //metodo main  
        public static void main(String[] args) {  
            Formulario22 fr= new Formulario22();  
            fr.setVisible(true);  
        }  
    }  
}
```

## Formulario:

```
package Cap_Ejercicio22;  
  
import java.awt.EventQueue;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
import javax.swing.JFrame;  
import javax.swing.JPanel;  
import javax.swing.border.EmptyBorder;  
  
import javax.swing.JLabel;  
import javax.swing.JOptionPane;  
import javax.swing.JTextField;
```

```
import javax.swing.JButton;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JEditorPane;
import javax.swing.JSeparator;

public class Formulario22 extends JFrame {

    private JPanel contentPane;
    private JTextField textNombre;
    private JTextField textSalario;
    private JTextField textHoras;
    private JTextField textField;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Formulario22 frame = new
Formulario22();
                    frame.setVisible(true);
                } catch (Exception e) {
```

```

        e.printStackTrace();
    }
}

});
}

/**
 * Create the frame.
 */
public Formulario22() {
    this.setTitle("Ejercicio 22");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 615, 332);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(204, 204,
204));
    contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    JLabel lblNewLabel = new JLabel("Ingrese
nombre");
    lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 14));

```

```
lblNewLabel.setBounds(41, 114, 170, 14);  
contentPane.add(lblNewLabel);
```

```
JLabel lblSalarioPorHora = new JLabel("Ingrese  
salario por hora");
```

```
lblSalarioPorHora.setFont(new Font("Verdana",  
Font.PLAIN, 14));
```

```
lblSalarioPorHora.setBounds(41, 139, 207, 14);  
contentPane.add(lblSalarioPorHora);
```

```
JLabel lblCantidadHorasTrabajadas = new  
JLabel("Ingrese cantidad horas trabajadas al mes");
```

```
lblCantidadHorasTrabajadas.setFont(new  
Font("Verdana", Font.PLAIN, 14));
```

```
lblCantidadHorasTrabajadas.setBounds(40, 164,  
330, 14);  
contentPane.add(lblCantidadHorasTrabajadas);
```

```
textNombre = new JTextField();  
textNombre.setBounds(390, 101, 162, 20);  
contentPane.add(textNombre);  
textNombre.setColumns(10);
```

```
textSalario = new JTextField();  
textSalario.setColumns(10);  
textSalario.setBounds(390, 132, 162, 20);  
contentPane.add(textSalario);
```

```

textHoras = new JTextField();
textHoras.setColumns(10);
textHoras.setBounds(390, 163, 162, 20);
contentPane.add(textHoras);

textField = new JTextField();
textField.setBounds(29, 223, 365, 20);
contentPane.add(textField);
textField.setColumns(10);

JButton btnCalcular = new JButton("Calcular");
btnCalcular.addActionListener(new
ActionListener() {
    public void actionPerformed(ActionEvent e)
    {
        //CALCULO DEL SALARIO
        try {
            double salario;
            String nombre;
            double horas;

            nombre = textNombre.getText();
            salario =
Double.parseDouble(textSalario.getText());

```

```

        horas =
Double.parseDouble(textHoras.getText());

        if
(Metodos22.Calcular_salario(salario,horas)== 100) {
            textField.setText(nombre);
        }
        else {
            textField.setText(nombre+" su
salario es "+Metodos22.Calcular_salario(salario,horas));;
        }
    }
    catch(NumberFormatException ex) {
        //EXCEPCION VALORES NO
NÚMERICOS

        JOptionPane.showMessageDialog(null, "Los parámetros
ingresados no son numéricos");

    }

}

});

btnCalcular.setBounds(69, 189, 89, 23);
contentPane.add(btnCalcular);

```

```

        JButton btnBorrar = new JButton("Borrar");
        btnBorrar.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                //CONFIGURA EL BOTON BORRAR
                textField.setText("");
                textSalario.setText("");
                textNombre.setText("");
                textHoras.setText("");
            }
        });
        btnBorrar.setBounds(196, 189, 89, 23);
        contentPane.add(btnBorrar);

        JEditorPane dtrpnEsteProgramaLe = new
        JEditorPane();

        dtrpnEsteProgramaLe.setText("Este programa le
        permitirá calcular el salario de un empleado siempre y
        cuando sea mayor a 450.000");

        dtrpnEsteProgramaLe.setForeground(Color.BLACK);
        dtrpnEsteProgramaLe.setFont(new Font("Sitka
        Subheading", Font.BOLD | Font.ITALIC, 14));
        dtrpnEsteProgramaLe.setBackground(new Color(204,
        204, 204));
        dtrpnEsteProgramaLe.setBounds(19, 11, 533, 42);

```



```

        contentPane.add(dtrpnEsteProgramaLe);

        JSeparator separator = new JSeparator();
        separator.setBackground(new Color(51, 102, 51));
        separator.setBounds(848, 64, -818, -21);
        contentPane.add(separator);

        JSeparator separator_1 = new JSeparator();
        separator_1.setBackground(new Color(51, 102,
51));
        separator_1.setBounds(29, 63, 533, 14);
        contentPane.add(separator_1);
    }
}

```

## Métodos:

```

package Cap_Ejercicio22;

public class Metodos22 {
    //calcula el salario del trabajador en caso de ser
    mayor

    public static double Calcular_salario(double
salario, double horas) {

```

```
double pago;
pago = salario*horas;
if (pago > 450000) {

    return pago;
}else {
    pago = 100;
    return pago;

}

}
```

## EJERCICIO #23:

Ejercicio 23

Este programa le permitirá hallar las posibles soluciones de una ecuación de segundo grado (de la forma  $ax^2+bx+c$ ). Para ello ingrese los valores A,B,C y pulse en "Calcular".

Ingrese valor de A

Ingrese valor de B

Ingrese valor de C

RESULTADO

Programa main:

```
package Cap4_Ejercicio23;
```

```
public class Ejercicio_23 {  
    //metodo main  
    public static void main(String[] args) {  
        Formulario_23 fr= new Formulario_23();  
        fr.setVisible(true);  
    }  
}
```

## Formulario:

```
package Cap4_Ejercicio23;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.Font;
import javax.swing.JEditorPane;
import javax.swing.JSeparator;
import java.awt.Color;

public class Formulario_23 extends JFrame {

    private JPanel contentPane;
    private JTextField txtA;
    private JTextField txtB;
```

```

private JTextField txtC;
private JTextField txtResultado;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Formulario_23 frame = new
Formulario_23();

                frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the frame.
 */
public Formulario_23() {
    this.setTitle("Ejercicio 23");
}

```

```
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 700, 285);
        contentPane = new JPanel();
        contentPane.setBackground(new Color(204, 204,
204));
        contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));
```

```
        setContentPane(contentPane);
        contentPane.setLayout(null);
```

```
        JLabel lblNewLabel = new JLabel("Ingrese valor
de A");
```

```
        lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel.setBounds(10, 82, 124, 14);
        contentPane.add(lblNewLabel);
```

```
        JLabel lblIngreseValorDe = new JLabel("Ingrese
valor de B");
```

```
        lblIngreseValorDe.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblIngreseValorDe.setBounds(271, 82, 124, 14);
        contentPane.add(lblIngreseValorDe);
```

```
        JLabel lblIngreseValorDe_1 = new JLabel("Ingrese
valor de C");
```

```
        lblIngreseValorDe_1.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblIngreseValorDe_1.setBounds(537, 82, 124, 14);
        contentPane.add(lblIngreseValorDe_1);

        txtA = new JTextField();
        txtA.setBounds(30, 99, 86, 20);
        contentPane.add(txtA);
        txtA.setColumns(10);

        txtB = new JTextField();
        txtB.setBounds(286, 99, 86, 20);
        contentPane.add(txtB);
        txtB.setColumns(10);

        txtC = new JTextField();
        txtC.setBounds(557, 99, 86, 20);
        contentPane.add(txtC);
        txtC.setColumns(10);

        JButton btnCalcular = new JButton("Calcular");
        btnCalcular.setFont(new Font("Verdana",
Font.PLAIN, 11));
        btnCalcular.addActionListener(new
ActionListener() {
```

```

        public void actionPerformed(ActionEvent e)
    {
        double A, B, C;
        double Respuesta1, Respuesta2;
        try {

            //CALCULO DE ECUACION CUADRATICA.
            A=Double.parseDouble(txtA.getText());
            B=Double.parseDouble(txtB.getText());
            C=Double.parseDouble(txtC.getText());
            double determinante= Math.pow(B, 2) -
(4 * A * C);

            System.out.println(determinante);

            if(determinante<0) {
                txtResultado.setText("Las
soluciones para la ecuación cuadratica son soluciones
complejas");
            }
            else if(determinante==0) {

txtResultado.setText(String.valueOf("X=
"+Métodos.Calcular_Cuadratica_determinante_cero(A,B,
determinante)));

```



```

        }
        else {

Respuesta1=Métodos.Calcular_Cuadratica_Positiva(A,B,
determinante);

Respuesta2=Métodos.Calcular_Cuadratica_Negativa(A,B,
determinante);

                                txtResultado.setText("X1=
"+String.valueOf(Respuesta1)+"      X2=
"+String.valueOf(Respuesta2));
        }
        }catch(NumberFormatException ex) {

                                //EXCEPCION VALORES NO
NÚMERICOS

                                JOptionPane.showMessageDialog(null, "Rellene todos
los campos o utilice un formato valido");

        }

    }

});

btnCalcular.setBounds(248, 147, 89, 23);
contentPane.add(btnCalcular);

JButton btnBorrar = new JButton("Borrar");

```

```
        btnBorrar.setFont(new Font("Verdana",
Font.PLAIN, 11));
        btnBorrar.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        txtA.setText("");
        txtB.setText("");
        txtC.setText("");
        txtResultado.setText("");

    }
});
        btnBorrar.setBounds(357, 147, 89, 23);
        contentPane.add(btnBorrar);

        JLabel lblNewLabel_10 = new JLabel("RESULTADO");
        lblNewLabel_10.setFont(new Font("Arial Rounded
MT Bold", Font.PLAIN, 15));
        lblNewLabel_10.setBounds(30, 186, 156, 14);
        contentPane.add(lblNewLabel_10);

        txtResultado = new JTextField();
        txtResultado.setBounds(167, 185, 407, 20);
        contentPane.add(txtResultado);
        txtResultado.setColumns(10);
```

```
JEditorPane dtrpnEsteProgramaLe = new
JEditorPane();

dtrpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 12));

dtrpnEsteProgramaLe.setBackground(new Color(204,
204, 204));

dtrpnEsteProgramaLe.setText("Este programa le
permitirá hallar las posibles soluciones de una ecuación
de segundo grado (de la forma  $ax^2+bx+c$ ).");

dtrpnEsteProgramaLe.setBounds(10, 11, 664, 20);
contentPane.add(dtrpnEsteProgramaLe);


JEditorPane dtrpnParaElloIngrese = new
JEditorPane();

dtrpnParaElloIngrese.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 12));

dtrpnParaElloIngrese.setBackground(new
Color(204, 204, 204));

dtrpnParaElloIngrese.setText("Para ello ingrese
los valores A,B,C y pulse en \"Calcular\".");

dtrpnParaElloIngrese.setBounds(10, 29, 479, 20);
contentPane.add(dtrpnParaElloIngrese);


JSeparator separator = new JSeparator();
separator.setBackground(new Color(0, 128, 0));
separator.setBounds(10, 222, 664, 13);
contentPane.add(separator);
```

```
    }  
}
```

## Métodos:

```
package Cap4_Ejercicio23;
```

```
import java.text.DecimalFormat;
```

```
public class Métodos {  
    public static String  
    Calcular_Cuadratica_determinante_cero(double A,double  
    B,double determinante) {  
        //CALCULA LA RAIZ CUADRATICA CUANDO EL  
        DETERMINANTE ES CERO  
        DecimalFormat df = new DecimalFormat("#.0000");  
        return df.format(-B/ (2 * A));  
    }  
  
    public static double  
    Calcular_Cuadratica_Positiva(double A,double B,double  
    determinante) {  
        //CALCULA LA RAIZ CUADRATICA CON DETERMINANTE  
        MAYOR QUE CERO  
        DecimalFormat df = new DecimalFormat("#.00");  
        return (-B + Math.sqrt(determinante)) / (2 * A);  
    }  
}
```

```

    }

    public static double
    Calcular_Cuadratica_Negativa(double A,double B,double
    determinante) {

        //CALCULA LA RAIZ CUADRATICA CON DETERMINANTE
        MAYOR QUE CERO

        return (-B - Math.sqrt(determinante)) / (2 * A);

    }
}

```

## EJERCICIO #40:

Ejercicio 40

*Este programa le permitirá calcular la raíz cuadrada, la potencia cuadrada y la potencia al cubo de un grupo de 7 números enteros positivos. Escriba los números con los que desee trabajar y pulse en "Calcular".*

	Introduzca Número 1	Introduzca Número 2	Introduzca Número 3	Introduzca Número 4	Introduzca Número 5	Introduzca Número 6	Introduzca Número 7
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<b>RESULTADOS</b>							
Raíz cuadrada	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Potencia cuadrada	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Potencia cúbica	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="Calcular"/> <input type="button" value="Borrar"/>							

```
package Cap4_Ejercicio40;

public class Ejercicio40 {
    //metodo main
    public static void main(String[] args) {
        formulario_40 fr= new formulario_40();
        fr.setVisible(true);
    }
}
```

## Formulario:

```
package Cap4_Ejercicio40;

import java.awt.EventQueue;
import javax.swing.JOptionPane;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JList;
```

```
import javax.swing.JToggleButton;
import javax.swing.JSpinner;
import javax.swing.JLabel;
import javax.swing.JTextPane;
import javax.swing.JTextField;
import javax.swing.JSeparator;
import java.awt.Color;
import javax.swing.JSlider;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.util.FormatterClosedException;
import java.awt.event.ActionEvent;
import java.awt.Font;

public class formulario_40 extends JFrame {
    private JPanel contentPane;
    private JTextField textN3;
    private JTextField textN1;
    private JTextField textN2;
    private JTextField textN4;
    private JTextField textN5;
    private JTextField textN6;
    private JTextField textN7;
    private JSeparator separator;
    private JLabel lblNewLabel_7;
```

```
private JLabel lblNewLabel_8;  
private JLabel lblNewLabel_9;  
private JTextField textR1;  
private JTextField textCua1;  
private JTextField textCu1;  
private JTextField textR2;  
private JTextField textCua2;  
private JTextField textCu2;  
private JTextField textR3;  
private JTextField textCua3;  
private JTextField textCu3;  
private JTextField textR4;  
private JTextField textCua4;  
private JTextField textCu4;  
private JTextField textR5;  
private JTextField textCua5;  
private JTextField textCu5;  
private JTextField textR6;  
private JTextField textCua6;  
private JTextField textCu6;  
private JTextField textR7;  
private JTextField textCua7;  
private JTextField textCu7;  
private JTextPane txtpnEscribaLosNmeros;  
private JLabel lblNewLabel_10;
```



```

private JLabel lblNewLabel_11;
private JLabel lblNewLabel_12;
private JLabel lblNewLabel_13;
private JLabel lblNewLabel_14;
private JLabel lblNewLabel_15;
private JLabel lblNewLabel_16;
private JLabel lblNewLabel_17;
private JLabel lblNewLabel_1;
private JLabel lblNewLabel_2;
private JLabel lblNewLabel_3;
private JLabel lblNewLabel_4;
private JLabel lblNewLabel_5;
private JSeparator separator_1;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                formulario_40 frame = new
formulario_40();
                frame.setVisible(true);
            } catch (Exception e) {

```

```

        e.printStackTrace();
    }
}

});
}

/**
 * Create the frame.
 */
public formulario_40() {
    this.setTitle("Ejercicio 40 ");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 1044, 494);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(204, 204,
204));
    contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    JLabel lblNewLabel_6 = new JLabel("Introduzca
");
    lblNewLabel_6.setFont(new Font("Verdana",
Font.PLAIN, 11));

```

```
lblNewLabel_6.setBounds(925, 65, 83, 14);
contentPane.add(lblNewLabel_6);

JTextPane txtpnEsteProgramaLe = new JTextPane();
txtpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 12));
txtpnEsteProgramaLe.setBackground(new Color(204,
204, 204));
txtpnEsteProgramaLe.setText("Este programa le
permitirá calcular la raíz cuadrada, la potencia cuadrada
y la potencia al cubo de un grupo de 7 números enteros
positivos.");
txtpnEsteProgramaLe.setBounds(96, 11, 828, 20);
contentPane.add(txtpnEsteProgramaLe);

textN3 = new JTextField();
textN3.setColumns(10);
textN3.setBounds(417, 90, 86, 20);
contentPane.add(textN3);

JLabel lblNewLabel = new JLabel("Introduzca ");
lblNewLabel.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel.setBounds(146, 65, 106, 14);
contentPane.add(lblNewLabel);

textN1 = new JTextField();
```

```
textN1.setColumns(10);  
textN1.setBounds(146, 90, 86, 20);  
contentPane.add(textN1);
```

```
textN2 = new JTextField();  
textN2.setColumns(10);  
textN2.setBounds(276, 90, 86, 20);  
contentPane.add(textN2);
```

```
textN4 = new JTextField();  
textN4.setColumns(10);  
textN4.setBounds(547, 90, 86, 20);  
contentPane.add(textN4);
```

```
textN5 = new JTextField();  
textN5.setColumns(10);  
textN5.setBounds(679, 90, 86, 20);  
contentPane.add(textN5);
```

```
textN6 = new JTextField();  
textN6.setColumns(10);  
textN6.setBounds(800, 90, 86, 20);  
contentPane.add(textN6);
```

```
textN7 = new JTextField();
```

```
textN7.setColumns(10);  
textN7.setBounds(922, 90, 86, 20);  
contentPane.add(textN7);
```

```
separator = new JSeparator();  
separator.setBackground(new Color(0, 128, 64));  
separator.setBounds(10, 133, 1010, 2);  
contentPane.add(separator);
```

```
lblNewLabel_7 = new JLabel("Raíz cuadrada");  
lblNewLabel_7.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_7.setBounds(10, 183, 126, 14);  
contentPane.add(lblNewLabel_7);
```

```
lblNewLabel_8 = new JLabel("Potencia cuadrada");  
lblNewLabel_8.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_8.setBounds(10, 247, 146, 14);  
contentPane.add(lblNewLabel_8);
```

```
lblNewLabel_9 = new JLabel("Potencia cúbica");  
lblNewLabel_9.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_9.setBounds(10, 310, 106, 14);  
contentPane.add(lblNewLabel_9);
```

```
textR1 = new JTextField();  
textR1.setColumns(10);  
textR1.setBounds(146, 181, 86, 20);  
contentPane.add(textR1);
```

```
textCua1 = new JTextField();  
textCua1.setColumns(10);  
textCua1.setBounds(146, 245, 86, 20);  
contentPane.add(textCua1);
```

```
textCu1 = new JTextField();  
textCu1.setColumns(10);  
textCu1.setBounds(146, 308, 86, 20);  
contentPane.add(textCu1);
```

```
textR2 = new JTextField();  
textR2.setColumns(10);  
textR2.setBounds(276, 180, 86, 20);  
contentPane.add(textR2);
```

```
textCua2 = new JTextField();  
textCua2.setColumns(10);  
textCua2.setBounds(276, 244, 86, 20);  
contentPane.add(textCua2);
```

```
textCu2 = new JTextField();  
textCu2.setColumns(10);  
textCu2.setBounds(276, 307, 86, 20);  
contentPane.add(textCu2);
```

```
textR3 = new JTextField();  
textR3.setColumns(10);  
textR3.setBounds(417, 180, 86, 20);  
contentPane.add(textR3);
```

```
textCua3 = new JTextField();  
textCua3.setColumns(10);  
textCua3.setBounds(417, 244, 86, 20);  
contentPane.add(textCua3);
```

```
textCu3 = new JTextField();  
textCu3.setColumns(10);  
textCu3.setBounds(417, 307, 86, 20);  
contentPane.add(textCu3);
```

```
textR4 = new JTextField();  
textR4.setColumns(10);  
textR4.setBounds(547, 180, 86, 20);  
contentPane.add(textR4);
```

```
textCua4 = new JTextField();  
textCua4.setColumns(10);  
textCua4.setBounds(547, 244, 86, 20);  
contentPane.add(textCua4);
```

```
textCu4 = new JTextField();  
textCu4.setColumns(10);  
textCu4.setBounds(547, 307, 86, 20);  
contentPane.add(textCu4);
```

```
textR5 = new JTextField();  
textR5.setColumns(10);  
textR5.setBounds(679, 180, 86, 20);  
contentPane.add(textR5);
```

```
textCua5 = new JTextField();  
textCua5.setColumns(10);  
textCua5.setBounds(679, 244, 86, 20);  
contentPane.add(textCua5);
```

```
textCu5 = new JTextField();  
textCu5.setColumns(10);  
textCu5.setBounds(679, 307, 86, 20);  
contentPane.add(textCu5);
```



```
textR6 = new JTextField();  
textR6.setColumns(10);  
textR6.setBounds(800, 180, 86, 20);  
contentPane.add(textR6);
```

```
textCua6 = new JTextField();  
textCua6.setColumns(10);  
textCua6.setBounds(800, 244, 86, 20);  
contentPane.add(textCua6);
```

```
textCu6 = new JTextField();  
textCu6.setColumns(10);  
textCu6.setBounds(800, 307, 86, 20);  
contentPane.add(textCu6);
```

```
textR7 = new JTextField();  
textR7.setColumns(10);  
textR7.setBounds(922, 180, 86, 20);  
contentPane.add(textR7);
```

```
textCua7 = new JTextField();  
textCua7.setColumns(10);  
textCua7.setBounds(922, 244, 86, 20);  
contentPane.add(textCua7);
```

```

        textCu7 = new JTextField();
        textCu7.setColumns(10);
        textCu7.setBounds(922, 307, 86, 20);
        contentPane.add(textCu7);

        JButton btnNewButton = new JButton("Calcular");
        btnNewButton.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
        {
            //CALCULO DE RAICES, CUADRADOS Y CUBOS.
            try {

                double numero;

                if(textN1.getText()!="") {

                    numero=Double.parseDouble(textN1.getText());

                    textR1.setText(String.valueOf(Métodos.Calcular_raices(numero)));

                    textCua1.setText(String.valueOf(Métodos.Calcular_cuadrado(numero)));

```

```
        textCu1.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
    }
```

```
        if(textN2.getText()!="") {
```

```
            numero=Double.parseDouble(textN2.getText());
```

```
            textR2.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
            textCua2.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
            textCu2.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
        }
```

```
        if(textN3.getText()!="") {
```

```
            numero=Double.parseDouble(textN3.getText());
```

```
            textR3.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
        textCua3.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
        textCu3.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
    }
```

```
        if(textN4.getText()!="") {
```

```
            numero=Double.parseDouble(textN4.getText());
```

```
            textR4.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
            textCua4.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
            textCu4.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
        }
```

```
        if(textN5.getText()!="") {
```

```
            numero=Double.parseDouble(textN5.getText());
```

```
            textR5.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
        textCua5.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
        textCu5.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
    }
```

```
        if(textN6.getText()!="") {
```

```
            numero=Double.parseDouble(textN6.getText());
```

```
            textR6.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
            textCua6.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
            textCu6.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
        }
```

```
        if(textN7.getText()!="") {
```

```
            numero=Double.parseDouble(textN7.getText());
```

```
            textR7.setText(String.valueOf(Métodos.Calcular_raice  
s(numero)));
```

```
        textCua7.setText(String.valueOf(Métodos.Calcular_cua  
drado(numero)));
```

```
        textCu7.setText(String.valueOf(Métodos.Calcular_cubo  
(numero)));
```

```
    }
```

```
    }catch(NumberFormatException ex) {
```

```
        //EXCEPCION VALORES NO NÚMERICOS
```

```
        JOptionPane.showMessageDialog(null,  
"Rellene todos los campos o ingrese un formato valido");
```

```
    }
```

```
    }
```

```
});
```

```
btnNewButton.setBounds(368, 376, 89, 23);
```

```
contentPane.add(btnNewButton);
```

```
JButton btnBorrar = new JButton("Borrar");
```

```
btnBorrar.addActionListener(new ActionListener()
```

```
{
```

```
    public void actionPerformed(ActionEvent e)
```

```
{
```

```
        //CONFIGURA EL BOTON BORRAR
```

```
        textR1.setText("");
```

```
        textCua1.setText("");
```

```
        textCu1.setText("");
```

```
textR2.setText("");  
textCua2.setText("");  
textCu2.setText("");
```

```
textR3.setText("");  
textCua3.setText("");  
textCu3.setText("");
```

```
textR4.setText("");  
textCua4.setText("");  
textCu4.setText("");
```

```
textR5.setText("");  
textCua5.setText("");  
textCu5.setText("");
```

```
textR6.setText("");  
textCua6.setText("");  
textCu6.setText("");
```

```
textR7.setText("");  
textCua7.setText("");  
textCu7.setText("");
```

```

        textN1.setText("");
        textN2.setText("");
        textN3.setText("");
        textN4.setText("");
        textN5.setText("");
        textN6.setText("");
        textN7.setText("");
    }
});
btnBorrar.setBounds(521, 376, 89, 23);
contentPane.add(btnBorrar);

txtpnEscribaLosNmeros = new JTextPane();
txtpnEscribaLosNmeros.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 12));
txtpnEscribaLosNmeros.setBackground(new
Color(204, 204, 204));
txtpnEscribaLosNmeros.setText("Escriba los
números con los que desee trabajar y pulse en
\"Calcular\".");
txtpnEscribaLosNmeros.setBounds(96, 26, 828,
20);

contentPane.add(txtpnEscribaLosNmeros);

lblNewLabel_10 = new JLabel("RESULTADOS");
lblNewLabel_10.setFont(new Font("Arial Rounded
MT Bold", Font.PLAIN, 15));

```



```
lblNewLabel_10.setBounds(454, 143, 156, 14);  
contentPane.add(lblNewLabel_10);
```

```
lblNewLabel_11 = new JLabel("Número 1");  
lblNewLabel_11.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_11.setBounds(149, 76, 83, 14);  
contentPane.add(lblNewLabel_11);
```

```
lblNewLabel_12 = new JLabel("Número 2");  
lblNewLabel_12.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_12.setBounds(276, 76, 93, 14);  
contentPane.add(lblNewLabel_12);
```

```
lblNewLabel_13 = new JLabel("Número 3");  
lblNewLabel_13.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_13.setBounds(417, 76, 83, 14);  
contentPane.add(lblNewLabel_13);
```

```
lblNewLabel_14 = new JLabel("Número 4");  
lblNewLabel_14.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
lblNewLabel_14.setBounds(547, 76, 93, 14);  
contentPane.add(lblNewLabel_14);
```

```
        lblNewLabel_15 = new JLabel("Número 5");  
        lblNewLabel_15.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
        lblNewLabel_15.setBounds(679, 76, 78, 14);  
        contentPane.add(lblNewLabel_15);
```

```
        lblNewLabel_16 = new JLabel("Número 6");  
        lblNewLabel_16.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
        lblNewLabel_16.setBounds(800, 76, 88, 14);  
        contentPane.add(lblNewLabel_16);
```

```
        lblNewLabel_17 = new JLabel("Número 7");  
        lblNewLabel_17.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
        lblNewLabel_17.setBounds(925, 76, 77, 14);  
        contentPane.add(lblNewLabel_17);
```

```
        lblNewLabel_1 = new JLabel("Introduzca ");  
        lblNewLabel_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
        lblNewLabel_1.setBounds(276, 65, 83, 14);  
        contentPane.add(lblNewLabel_1);
```

```
        lblNewLabel_2 = new JLabel("Introduzca ");
```

```
        lblNewLabel_2.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_2.setBounds(417, 65, 104, 14);
        contentPane.add(lblNewLabel_2);

        lblNewLabel_3 = new JLabel("Introduzca ");
        lblNewLabel_3.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_3.setBounds(547, 65, 83, 14);
        contentPane.add(lblNewLabel_3);

        lblNewLabel_4 = new JLabel("Introduzca ");
        lblNewLabel_4.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_4.setBounds(679, 65, 83, 14);
        contentPane.add(lblNewLabel_4);

        lblNewLabel_5 = new JLabel("Introduzca ");
        lblNewLabel_5.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_5.setBounds(800, 65, 83, 14);
        contentPane.add(lblNewLabel_5);

        separator_1 = new JSeparator();
        separator_1.setBackground(new Color(0, 128,
64));
        separator_1.setBounds(10, 426, 1010, 2);
```

```
        contentPane.add(separator_1);

    }

}
```

## Métodos:

```
package Cap4_Ejercicio40;

import java.text.DecimalFormat;

public class Métodos {
    //calcula las raiz cuadrada del numero
    public static String Calcular_raices(double numero){
        DecimalFormat df = new DecimalFormat("#.0000");
        return df.format(Math.sqrt(numero));

        //calcula el cuadrado del numero
    }
    public static double Calcular_cuadrado(double
numero){
        return Math.pow(numero,2);
    }
}
```

```

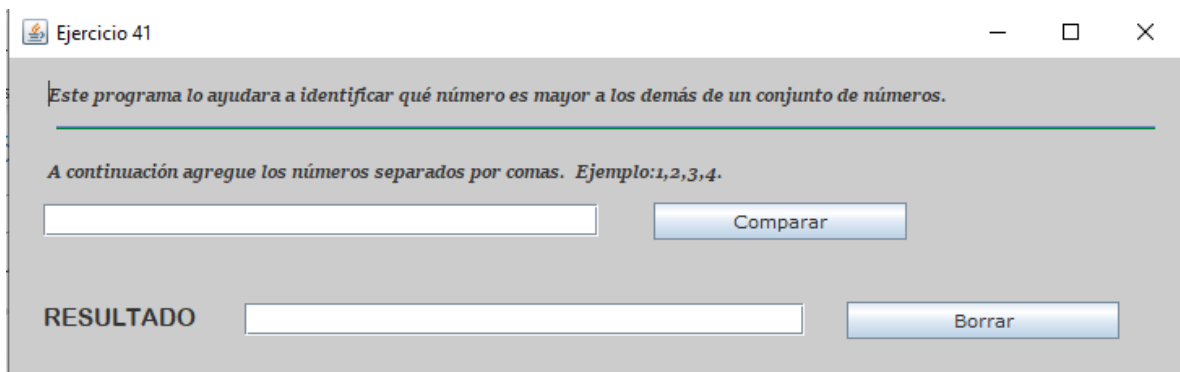
    }
    //clcula el cubo del numero
    public static double Calcular_cubo(double numero){

        return Math.pow(numero,3);

    }
}

```

## EJERCICIO #41:



**Ejercicio 41**

Este programa lo ayudara a identificar qué número es mayor a los demás de un conjunto de números.

---

A continuación agregue los números separados por comas. Ejemplo:1,2,3,4.

**RESULTADO**

## Programa main:

```
package Cap4_Ejercicio41;
```

```
public class Ejercicio41 {  
    //main  
    public static void main(String[] args) {  
        Formulario_41 fr= new Formulario_41();  
        fr.setVisible(true);  
        double a;  
    }  
  
}
```

## Formulario:

```
package Cap4_Ejercicio41;  
  
import java.awt.EventQueue;  
  
import javax.swing.JFrame;  
import javax.swing.JPanel;  
import javax.swing.border.EmptyBorder;  
import javax.swing.JLabel;  
import javax.swing.JOptionPane;  
import javax.swing.JTextField;
```

```
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JTextPane;
import javax.swing.JSeparator;
import java.awt.Color;
import java.awt.Font;

public class Formulario_41 extends JFrame {

    private JPanel contentPane;
    private JTextField textNros;
    protected double cosa;
    private JTextField textResultado;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {

            public void run() {

                try {
```

```
        Formulario_41 frame = new Formulario_41();
        frame.setVisible(true);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

});
}
```

```
/**
```

```
 * Create the frame.
```

```
 */
```

```
public Formulario_41() {
```

```
this.setTitle("Ejercicio 41");
```

```
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
    setBounds(100, 100, 739, 234);
```

```
    contentPane = new JPanel();
```

```
    contentPane.setBackground(new Color(204, 204, 204));
```

```
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
```

```
    setContentPane(contentPane);
```



```
contentPane.setLayout(null);
```

```
textNros = new JTextField();
```

```
textNros.setBounds(21, 90, 342, 20);
```

```
contentPane.add(textNros);
```

```
textNros.setColumns(10);
```

```
JButton btnComparar = new JButton("Comparar");
```

```
btnComparar.setFont(new Font("Verdana", Font.PLAIN, 11));
```

```
btnComparar.addActionListener(new ActionListener() {
```

```
    public void actionPerformed(ActionEvent e) {
```

```
        String numeros;
```

```
        try {
```

```
            numeros=textNros.getText();
```

```
            //CALCULO DEL NÚMERO MAYOR.
```

```
            textResultado.setText("El mayor número de los  
elementos introducidos es: "+metodos.Calcular_mayor_numero(numeros));
```

```
        }catch(NumberFormatException ex) {
```

```
            //EXCEPCION VALORES NO NUMERICOS
```

```
            JOptionPane.showMessageDialog(null,  
"Rellene todos los campos o utilice un formato valido");
```

```
}  
}
```

```
});
```

```
btnComparar.setBounds(397, 89, 156, 23);  
contentPane.add(btnComparar);
```

```
JTextPane txtpnEsteProgramaLo = new JTextPane();  
txtpnEsteProgramaLo.setFont(new Font("Sitka Subheading",  
Font.BOLD | Font.ITALIC, 12));  
txtpnEsteProgramaLo.setBackground(new Color(204, 204, 204));  
txtpnEsteProgramaLo.setText("Este programa lo ayudara a  
identificar qué número es mayor a los demás de un conjunto de números.");  
txtpnEsteProgramaLo.setBounds(21, 11, 618, 20);  
contentPane.add(txtpnEsteProgramaLo);
```

```
JButton btnBorrar = new JButton("Borrar");  
btnBorrar.setFont(new Font("Verdana", Font.PLAIN, 11));  
btnBorrar.addActionListener(new ActionListener() {
```

```
public void actionPerformed(ActionEvent e) {
```

```
        textNros.setText("");
        textResultado.setText("");

    }

});

btnBorrar.setBounds(516, 150, 168, 23);
contentPane.add(btnBorrar);

JSeparator separator = new JSeparator();
separator.setBackground(new Color(0, 128, 64));
separator.setBounds(10, 55, 982, -4);
contentPane.add(separator);

JTextPane txtpnAcontinuacinAgregueLos = new JTextPane();
txtpnAcontinuacinAgregueLos.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 12));
txtpnAcontinuacinAgregueLos.setBackground(new Color(204,
204, 204));

txtpnAcontinuacinAgregueLos.setText("A continuación agregue
los números separados por comas. Ejemplo:1,2,3,4.");

txtpnAcontinuacinAgregueLos.setBounds(21, 58, 466, 20);
contentPane.add(txtpnAcontinuacinAgregueLos);

JSeparator separator_1 = new JSeparator();
```

```

        separator_1.setForeground(new Color(46, 139, 87));
        separator_1.setBackground(new Color(0, 100, 0));
        separator_1.setBounds(44, 42, 899, -2);
        contentPane.add(separator_1);

        JSeparator separator_2 = new JSeparator();
        separator_2.setBackground(new Color(0, 128, 64));
        separator_2.setBounds(29, 42, 677, 9);
        contentPane.add(separator_2);

        JLabel lblNewLabel_10 = new JLabel("RESULTADO");
        lblNewLabel_10.setFont(new Font("Arial Rounded MT Bold",
Font.PLAIN, 15));
        lblNewLabel_10.setBounds(21, 152, 156, 14);
        contentPane.add(lblNewLabel_10);

        textResultado = new JTextField();
        textResultado.setBounds(145, 151, 345, 20);
        contentPane.add(textResultado);
        textResultado.setColumns(10);
    }
}

```

**Métodos:**

```

package Cap4_Ejercicio41;

public class metodos {
    //metodo que separa cadena de string y busca el
    mayor

    public static double Calcular_mayor_numero(String
    numeros) {
        double datoArray,cont=0;
        String split[]=numeros.split(",");
        for (int i=0;i<split.length;i++) {
            datoArray=Double.parseDouble(split[i]);
            if(datoArray>cont) {


                cont=datoArray;

            }//fin si
        }//fin for

        return cont;
    }
}

```

## PARTE 2: FIGURAS GEOMÉTRICAS

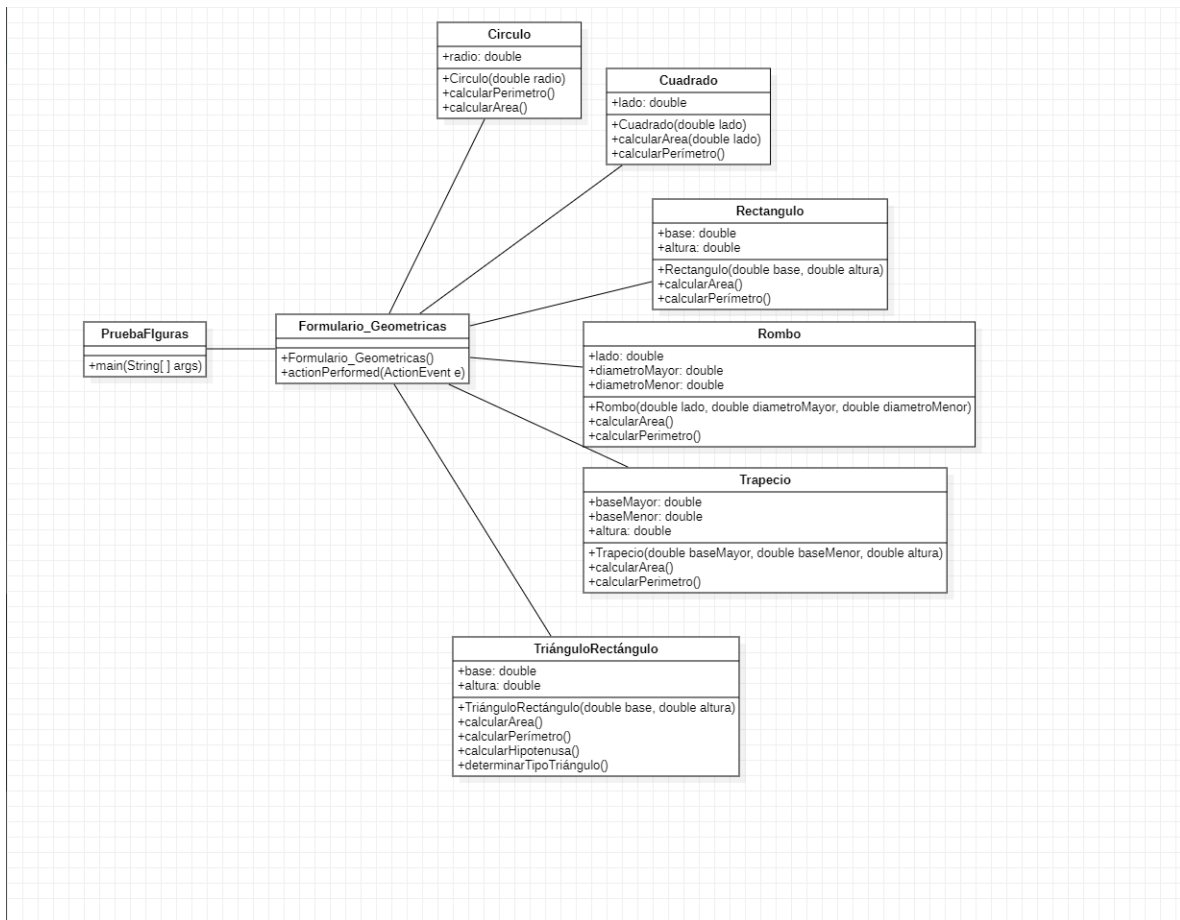
 Figuras geométricas

— □ ×

*Este programa le permitirá calcular el área y el perímetro de distintas figuras geométricas.  
Adicionalmente para el triángulo rectángulo podras calcular su hipotenusa e identificar su tipo de triángulo.*

Círculo	Ingrese el radio del círculo en centímetros <input type="text"/>	<input type="button" value="Calcular"/>	Area del círculo <input type="text"/>	Perímetro del círculo <input type="text"/>	<input type="button" value="Borrar"/>
Rectángulo	Ingrese base del rectángulo <input type="text"/> Ingrese altura del rectángulo <input type="text"/>	<input type="button" value="Calcular"/>	Area del rectángulo <input type="text"/>	Perímetro del rectángulo <input type="text"/>	<input type="button" value="Borrar"/>
Cuadrado	Ingrese longitud de los lados <input type="text"/>	<input type="button" value="Calcular"/>	Area del caudrado <input type="text"/>	Perímetro del cuadrado <input type="text"/>	<input type="button" value="Borrar"/>
Triángulo rectángulo	Ingrese base del triángulo <input type="text"/> Ingrese altura del triángulo <input type="text"/>	<input type="button" value="Calcular"/>	Hipotenusa del triángulo <input type="text"/> Area del triángulo <input type="text"/>	Perímetro del triángulo <input type="text"/> Tipo de triángulo <input type="text"/>	<input type="button" value="Borrar"/>
Rombo	Ingrese diámetro menor del rombo <input type="text"/> Ingrese diámetro mayor del rombo <input type="text"/> Ingrese lado del rombo <input type="text"/>	<input type="button" value="Calcular"/>	Area del rombo <input type="text"/>	Perímetro del rombo <input type="text"/>	<input type="button" value="Borrar"/>
Trapezio	Ingrese base mayor <input type="text"/> Ingrese base menor <input type="text"/> Ingrese altura <input type="text"/>	<input type="button" value="Calcular"/>	Area del trapezio <input type="text"/>	Perímetro del trapezio <input type="text"/>	<input type="button" value="Borrar"/>

## DIAGRAMA DE CLASES:



## Programa main:

```
package Figuras_Geometricas;
```

```

public class PruebaFiguras {
    //metodo main
    public static void main(String[] args) {
        Formulario_Geometricas fr=new
Formulario_Geometricas();
        fr.setVisible(true);

    }

}

```

## Circulo:

```

package Figuras_Geometricas;

/**
 * Constructor de la clase Circulo
 *
 * @param radio Parámetro que define el radio de un
círculo
 */
public class Circulo {
    double radio;
    Circulo(double radio){
        this.radio=radio;
    }
}

```



```

    }/**
    * Método que calcula área del círculo
    *
    * @return Área de un Circulo
    */

double calcularArea() {
    return Math.PI*Math.pow(radio, 2);

} /**
* Método que calcula el perímetro de un círculo
* @return Perímetro de un Circulo
*/
double calcularPerimetro() {
    return 2*Math.PI*radio;

}

}

```

**Cuadrado:**

```
package Figuras_Geometricas;
```

```
/**
 * Constructor de la clase Cuadrado
 * @param lado Parámetro que define la longitud de la
base de un
 * cuadrado
 */
public class Cuadrado {
double lado;

    public Cuadrado(double lado) {
        this.lado = lado;
    }

    /**
     * Método que calcula el área de un cuadrado
     * @return Área de un Cuadrado
     */
    double calcularArea() {
        return lado*lado;
    }

    /**
     * Método que calcula el perímetro de un cuadrado
     * @return Perímetro de un cuadrado
     */
    double calcularPerímetro() {
        return (4*lado);
    }
}
```

```
}
```

## Rectángulo:

```
package Figuras_Geometricas;
```

```
public class Rectangulo {
```

```
    double base;
```

```
    double altura;
```

```
    /**
```

```
    * Constructor de la clase Rectangulo
```

```
    * @param base Parámetro que define la base de un  
rectángulo
```

```
    * @param altura Parámetro que define la altura de un  
rectángulo
```

```
    */
```

```
    Rectangulo(double base, double altura) {
```

```
        this.base = base;
```

```
        this.altura = altura;
```

```
    /**
```

```
    * Método que calcula el área de un rectángulo
```

```
    * @return Área de un rectángulo
```

```
    */
```

```

    }
    double calcularArea() {
        return base * altura;
    }
    /**
     * Método que calcula el perímetro de un rectángulo
     * @return Perímetro de un rectángulo
     */
    double calcularPerímetro() {
        return (2 * base) + (2 * altura);
    }
}

```

## Rombo:

```

package Figuras_Geometricas;

public class Rombo {
    double lado;
    double diametroMayor;
    double diametroMenor;
    /**
     * Constructor de la clase Rombo
     *

```

```
    * @param lado          Parámetro que define el lado  
    de un rombo
```

```
    * @param diametroMayor Parámetro que define el  
    diámetro mayor de un rombo
```

```
    * @param diametroMenor Parámetro que define el  
    diámetro menor de un rombo
```

```
    */
```

```
    Rombo(double lado, double diametroMayor, double  
    diametroMenor) {
```

```
        this.lado = lado;
```

```
        this.diametroMayor = diametroMayor;
```

```
        this.diametroMenor = diametroMenor;
```

```
    }
```

```
    /**
```

```
    * Método que calcula el área de un rombo
```

```
    * @return Área de un rombo
```

```
    */
```

```
    double calcularArea() {
```

```
        return (diametroMayor * diametroMenor) / 2.0;
```

```
    }
```

```
    /**
```

```
    * Método que calcula el perímetro de un rectángulo
```

```
    * @return Perímetro de un rectángulo
```

```
    */
```

```
        double calcularPerimetro() {  
            return (4*lado);  
        }  
  
    }  
  
}
```

## Trapecio:

```
package Figuras_Geometricas;  
  
public class Trapecio {  
    double baseMayor;  
    double baseMenor;  
    double altura;  
    /**  
     * Constructor de la clase Rombo  
     *  
     * @param baseMayor Parámetro que define la base  
mayor de un trapecio  
     * @param baseMenor Parámetro que define la base  
menor de un trapecio  
     * @param altura Parámetro que define la altura de un  
trapecio  
     */  
    Trapecio(double baseMayor, double baseMenor, double  
altura) {
```

```

        this.baseMayor = baseMayor;
        this.baseMenor = baseMenor;
        this.altura = altura;
    }
    /**
     * Método que calcula el área de un trapecio
     * @return Área de un trapecio
     */
    double calcularArea() {
        return ((baseMayor + baseMenor) / 2.0)*altura;
    }
    /**
     * Método que calcula el perímetro de un trapecio
     * @return Perímetro de un trapecio
     */
    double calcularPerimetro() {
        return baseMenor + baseMayor +
2*Math.sqrt(Math.pow((baseMayor-baseMenor)/2.0,2) +
Math.pow(altura,2)) ;
    }

}

```

**Triángulo rectángulo:**

```

package Figuras_Geometricas;

```

```

/**
 * Constructor de la clase TriánguloRectángulo
 *
 * @param base    Parámetro que define la base de un
triángulo
 *
 *                rectángulo
 * @param altura Parámetro que define la altura de un
triángulo
 *
 *                rectángulo
 */
public class TriánguloRectángulo {
    double base;
    double altura;

    TriánguloRectángulo(double base, double altura) {
        this.base = base;
        this.altura = altura;
    }
}

/**
 * Método que calcula el área de un triángulo rectángulo
 *
 * @return Área de un triángulo rectángulo
 */

```



```

double calcularArea() {
    return (base * altura / 2);
}

/**
 * Método que calcula el perímetro de un triángulo
rectángulo
 * @return Perímetro de un triángulo rectángulo
 */
double calcularPerímetro() {
    return (base + altura + calcularHipotenusa());
}

/**
 * Método que calcula la hipotenusa de un triángulo
rectángulo
 * @return Hipotenusa de un triángulo rectángulo
 */
double calcularHipotenusa() {
    return Math.pow(base*base + altura*altura, 0.5);
}

/**
 * Método que determina si un triángulo es:
 * - Equilatero: si sus tres lados son iguales
 * - Escaleno: si sus tres lados son todos diferentes
 * - Escaleno: si dos de sus lados son iguales y el
otro es diferente de
 * los demás

```

```

    */
    int determinarTipoTriángulo() {

        if ((base == altura) && (base ==
calcularHipotenusa()) && (altura== calcularHipotenusa()))
        {

            return 1; //retorna tipo de triangulo
equilatero

        }

        else if((base != altura) && (base !=
calcularHipotenusa()) &&(altura !=
calcularHipotenusa())){

            return 2;      //retorna tipo de triangulo
escaleno

        }

        else {

            return 3; //retorna tipo de triangulo isoceles

        }

    }

}

```

## FORMULARIO:

```
package Figuras_Geometricas;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JSeparator;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JEditorPane;
import java.awt.Color;
import java.awt.Font;
import javax.swing.JTextField;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Formulario_Geometricas extends JFrame {

    private JPanel contentPane;
    private JTextField textRcirculo;
```

```
private JTextField textAcirculo;
private JTextField textPcirculo;
private JTextField textBrectangulo;
private JTextField textHrectangulo;
private JTextField textArectangulo;
private JTextField textPrectangulo;
private JTextField textLcuadrado;
private JTextField textAcuadrado;
private JTextField textPcuadrado;
private JTextField textHtriangulo;
private JTextField textBtriangulo;
private JTextField textHiptriangulo;
private JTextField textPtriangulo;
private JTextField textAtriangulo;
private JTextField textDMenor_rombo;
private JTextField textDMayor_rombo;
private JTextField textArombo;
private JTextField textPrombo;
private JTextField textBMayor_trapecio;
private JTextField textBMenor_trapecio;
private JTextField textHtrapecio;
private JTextField textAtrapecio;
private JTextField textPtrapecio;
private JTextField textTipotriangulo;
private JTextField textLrombo;
```

```

/**
 * Launch the application.
 */
/**
 * Create the frame.
 */
public Formulario_Geometricas() {
    this.setTitle("Figuras geométricas");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 1274, 957);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(204, 204,
204));
    contentPane.setBorder(new EmptyBorder(5, 5, 5,
5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    JSeparator separator = new JSeparator();
    separator.setBounds(10, 61, 1238, 11);
    contentPane.add(separator);

    JSeparator separator_1 = new JSeparator();
    separator_1.setBounds(10, 185, 1238, 11);

```

```
contentPane.add(separator_1);
```

```
JSeparator separator_2 = new JSeparator();  
separator_2.setBounds(10, 315, 1238, 11);  
contentPane.add(separator_2);
```

```
JSeparator separator_3 = new JSeparator();  
separator_3.setBounds(10, 452, 1238, 11);  
contentPane.add(separator_3);
```

```
JSeparator separator_4 = new JSeparator();  
separator_4.setBounds(10, 595, 1238, 11);  
contentPane.add(separator_4);
```

```
JSeparator separator_5 = new JSeparator();  
separator_5.setBounds(10, 759, 1238, 11);  
contentPane.add(separator_5);
```

```
JLabel lblNewLabel = new JLabel("Circulo\r\n");  
lblNewLabel.setFont(new Font("Verdana",  
Font.PLAIN, 16));  
lblNewLabel.setBounds(20, 117, 111, 14);  
contentPane.add(lblNewLabel);
```

```
JLabel lblNewLabel_1 = new  
JLabel("Rectángulo\r\n");
```

```
        lblNewLabel_1.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_1.setBounds(21, 251, 211, 14);
        contentPane.add(lblNewLabel_1);

        JLabel lblNewLabel_2 = new JLabel("Cuadrado");
        lblNewLabel_2.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_2.setBounds(22, 355, 132, 45);
        contentPane.add(lblNewLabel_2);

        JLabel lblNewLabel_2_1 = new JLabel("Triángulo
rectángulo");
        lblNewLabel_2_1.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_2_1.setBounds(10, 514, 226, 14);
        contentPane.add(lblNewLabel_2_1);

        JLabel lblNewLabel_2_1_1 = new JLabel("Rombo");
        lblNewLabel_2_1_1.setFont(new Font("Verdana",
Font.PLAIN, 16));
        lblNewLabel_2_1_1.setBounds(20, 664, 111, 14);
        contentPane.add(lblNewLabel_2_1_1);

        JLabel lblNewLabel_3 = new JLabel("Trapeccio");
        lblNewLabel_3.setFont(new Font("Verdana",
Font.PLAIN, 16));
```

```
lblNewLabel_3.setBounds(10, 828, 182, 14);
contentPane.add(lblNewLabel_3);

JEditorPane dtrpnEsteProgramaLe = new
JEditorPane();

dtrpnEsteProgramaLe.setFont(new Font("Sitka
Subheading", Font.BOLD | Font.ITALIC, 14));

dtrpnEsteProgramaLe.setBackground(new Color(204,
204, 204));

dtrpnEsteProgramaLe.setText("Este programa le
permitirá calcular el área y el perimetro de distintas
figuras geométricas.\r\nAdicionalmente para el triángulo
rectangulo podras calcular su hipotenusa e identificar
su tipo de triángulo.");

dtrpnEsteProgramaLe.setBounds(60, 11, 1034, 39);
contentPane.add(dtrpnEsteProgramaLe);

JLabel lblNewLabel_4 = new JLabel("Ingresa el
radio del circulo en centimetros\r\n\r\n\r\n");

lblNewLabel_4.setFont(new Font("Verdana",
Font.PLAIN, 11));

lblNewLabel_4.setBounds(165, 94, 323, 14);
contentPane.add(lblNewLabel_4);

textRcirculo = new JTextField();
textRcirculo.setBounds(240, 117, 139, 20);
contentPane.add(textRcirculo);
textRcirculo.setColumns(10);
```



```

        JButton btnCcirculo = new
JButton("Calcular\r\n");

        btnCcirculo.addActionListener(new
ActionListener() {

            public void actionPerformed(ActionEvent e)
{

                //CALCULA PROPIEDADES CIRCULO
                try {
                    double
radio=Double.parseDouble(textRcirculo.getText());
                    Circulo micirculo=new Circulo(radio);

                    textAcirculo.setText(String.valueOf(micirculo.calcul
arArea()));

                    textPcirculo.setText(String.valueOf(micirculo.calcul
arPerimetro()));

                }catch(NumberFormatException ex) {
                    //EXCEPCION VALORES NO NÚMERICOS

                    JOptionPane.showMessageDialog(null, "Rellene todos
los campos o ingrese un formato valido");

                }

```

```
        }  
    });  
    btnCcirculo.setBounds(511, 116, 89, 23);  
    contentPane.add(btnCcirculo);  
  
    JLabel lblNewLabel_5 = new JLabel("Area del  
circulo\r\n");  
    lblNewLabel_5.setFont(new Font("Verdana",  
Font.PLAIN, 11));  
    lblNewLabel_5.setBounds(673, 94, 123, 14);  
    contentPane.add(lblNewLabel_5);  
  
    textAcirculo = new JTextField();  
    textAcirculo.setBounds(652, 117, 144, 20);  
    contentPane.add(textAcirculo);  
    textAcirculo.setColumns(10);  
  
    JLabel lblNewLabel_5_1 = new JLabel("Perimetro  
del circulo\r\n");  
    lblNewLabel_5_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_5_1.setBounds(910, 94, 166, 14);  
contentPane.add(lblNewLabel_5_1);
```

```
textPcirculo = new JTextField();  
textPcirculo.setBounds(910, 117, 132, 20);  
contentPane.add(textPcirculo);  
textPcirculo.setColumns(10);
```

```
JLabel lblNewLabel_4_1 = new JLabel("Ingrese  
base del rectángulo");
```

```
lblNewLabel_4_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_4_1.setBounds(212, 196, 323, 14);  
contentPane.add(lblNewLabel_4_1);
```

```
textBrectangulo = new JTextField();  
textBrectangulo.setColumns(10);  
textBrectangulo.setBounds(240, 221, 139, 20);  
contentPane.add(textBrectangulo);
```

```
JLabel lblNewLabel_4_1_1 = new JLabel("Ingrese  
altura del rectángulo");
```

```
lblNewLabel_4_1_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
lblNewLabel_4_1_1.setBounds(212, 253, 323, 14);  
contentPane.add(lblNewLabel_4_1_1);
```

```

        textHrectangulo = new JTextField();
        textHrectangulo.setColumns(10);
        textHrectangulo.setBounds(240, 278, 139, 20);
        contentPane.add(textHrectangulo);

        JButton btnNewCrectangulo = new
        JButton("Calcular\r\n");

        btnNewCrectangulo.addActionListener(new
        ActionListener() {

            public void actionPerformed(ActionEvent e)
            {

                try {

                    //CALCULA PROPIEDADES RECTANGULO

                    double Brectangulo=
                    Double.parseDouble(textBrectangulo.getText()),Hrectangulo
                    = Double.parseDouble(textHrectangulo.getText());

                    Rectangulo miRectangulo = new
                    Rectangulo(Brectangulo,Hrectangulo);

                    textArectangulo.setText(String.valueOf(miRectangulo.
                    calcularArea()));

                    textPrectangulo.setText(String.valueOf(miRectangulo.
                    calcularPerímetro()));

                }catch(NumberFormatException ex) {

                    //EXCEPCION VALORES NO NUMERICOS

```

```

        JOptionPane.showMessageDialog(null, "Rellene todos
los campos o ingrese un formato valido");

    }

}

});

btnNewCrectangulo.setBounds(511, 233, 89, 23);
contentPane.add(btnNewCrectangulo);

JLabel lblNewLabel_5_2 = new JLabel("Area del
rectángulo");

lblNewLabel_5_2.setFont(new Font("Verdana",
Font.PLAIN, 11));

lblNewLabel_5_2.setBounds(659, 207, 153, 14);
contentPane.add(lblNewLabel_5_2);

textArectangulo = new JTextField();
textArectangulo.setColumns(10);
textArectangulo.setBounds(652, 234, 144, 20);
contentPane.add(textArectangulo);

JLabel lblNewLabel_5_1_1 = new JLabel("Perimetro
del rectángulo");

lblNewLabel_5_1_1.setFont(new Font("Verdana",
Font.PLAIN, 11));

```

```

lblNewLabel_5_1_1.setBounds(897, 207, 166, 14);
contentPane.add(lblNewLabel_5_1_1);

textPrectangulo = new JTextField();
textPrectangulo.setColumns(10);
textPrectangulo.setBounds(910, 234, 132, 20);
contentPane.add(textPrectangulo);

JLabel lblNewLabel_4_1_2 = new JLabel("Ingrese
longitud de los lados");
lblNewLabel_4_1_2.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel_4_1_2.setBounds(212, 345, 323, 14);
contentPane.add(lblNewLabel_4_1_2);

textLcuadrado = new JTextField();
textLcuadrado.setColumns(10);
textLcuadrado.setBounds(240, 370, 139, 20);
contentPane.add(textLcuadrado);

JButton btnCcuadrado = new
JButton("Calcular\r\n");
btnCcuadrado.addActionListener(new
ActionListener() {
    public void actionPerformed(ActionEvent e)
{
    try {

```

```

        //CALCULA PROPIEDADES CUADRADO
        double
lado=Double.parseDouble(textLcuadrado.getText());
        Cuadrado micuadrado=new
Cuadrado(lado);

        textAcuadrado.setText(String.valueOf(micuadrado.calc
ularArea()));

        textPcuadrado.setText(String.valueOf(micuadrado.calc
ularPerímetro()));
    }catch(NumberFormatException ex) {
        //EXCEPCION VALORES NO NÚMERICOS
        JOptionPane.showMessageDialog(null,
"Rellene todos los campos o ingrese un formato valido");

    }

}

});
btnCcuadrado.setBounds(511, 369, 89, 23);
contentPane.add(btnCcuadrado);

JLabel lblNewLabel_5_2_1 = new JLabel("Area del
caudrado\r\n\r\n");

```

```
        lblNewLabel_5_2_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
        lblNewLabel_5_2_1.setBounds(658, 346, 123, 14);
```

```
        contentPane.add(lblNewLabel_5_2_1);
```

```
        JLabel lblNewLabel_5_1_1_1 = new  
JLabel("Perimetro del cuadrado");
```

```
        lblNewLabel_5_1_1_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```

```
        lblNewLabel_5_1_1_1.setBounds(897, 337, 166,  
14);
```

```
        contentPane.add(lblNewLabel_5_1_1_1);
```

```
        textAcuadrado = new JTextField();
```

```
        textAcuadrado.setColumns(10);
```

```
        textAcuadrado.setBounds(652, 370, 132, 20);
```

```
        contentPane.add(textAcuadrado);
```

```
        textPcuadrado = new JTextField();
```

```
        textPcuadrado.setColumns(10);
```

```
        textPcuadrado.setBounds(910, 370, 132, 20);
```

```
        contentPane.add(textPcuadrado);
```

```
        JLabel lblNewLabel_4_1_2_1 = new JLabel("Ingrese  
altura del triángulo");
```

```
        lblNewLabel_4_1_2_1.setFont(new Font("Verdana",  
Font.PLAIN, 11));
```



```

14);

        lblNewLabel_4_1_2_1.setBounds(229, 539, 323,

        contentPane.add(lblNewLabel_4_1_2_1);


        textHtriangulo = new JTextField();
        textHtriangulo.setColumns(10);
        textHtriangulo.setBounds(240, 564, 139, 20);
        contentPane.add(textHtriangulo);


        JLabel lblNewLabel_4_1_3 = new JLabel("Ingrese
base del triángulo");
        lblNewLabel_4_1_3.setFont(new Font("Verdana",
Font.PLAIN, 11));
        lblNewLabel_4_1_3.setBounds(229, 474, 323, 14);
        contentPane.add(lblNewLabel_4_1_3);


        textBtriangulo = new JTextField();
        textBtriangulo.setColumns(10);
        textBtriangulo.setBounds(240, 491, 139, 20);
        contentPane.add(textBtriangulo);


        JButton btnCtriangulo = new
JButton("Calcular\r\n");
        btnCtriangulo.addActionListener(new
ActionListener() {
            public void actionPerformed(ActionEvent e)
{

```

```

        try {
            //CALCULA PROPIEDADES TRIÁNGULO

            double Btriangulo=
Double.parseDouble(textBtriangulo.getText()),Htriangulo=
Double.parseDouble(textHtriangulo.getText());

            TriánguloRectángulo miTriangulo = new
TriánguloRectángulo(Btriangulo,Htriangulo);

            textAtriangulo.setText(String.valueOf(miTriangulo.ca
lcularArea()));

            textPtriangulo.setText(String.valueOf(miTriangulo.ca
lcularPerímetro()));

            textHiptriangulo.setText(String.valueOf(miTriangulo.
calcularHipotenusa()));

            if(miTriangulo.determinarTipoTriángulo()==1) {
                textTipotriangulo.setText("El
triángulo es equilatero");
            }
            else
            if(miTriangulo.determinarTipoTriángulo()==2) {
                textTipotriangulo.setText("El
triángulo es escaleno");
            }
            else{

```

```

        textTipotriangulo.setText("El
triángulo es isóceles");
    }
    }catch(NumberFormatException ex) {
        //EXCEPCION VALORES NO NUMERICOS

        JOptionPane.showMessageDialog(null, "Rellene todos
los campos o ingrese un formato valido");
    }
}

```

```

    }
});
btnCtriangulo.setBounds(511, 499, 89, 23);
contentPane.add(btnCtriangulo);

JLabel lblNewLabel_5_2_1_1 = new
JLabel("Hipotenusa del triángulo");
    lblNewLabel_5_2_1_1.setFont(new Font("Verdana",
Font.PLAIN, 11));
    lblNewLabel_5_2_1_1.setBounds(652, 474, 173,
14);
contentPane.add(lblNewLabel_5_2_1_1);

textHiptriangulo = new JTextField();

```

```
textHiptriangulo.setColumns(10);
textHiptriangulo.setBounds(652, 491, 132, 20);
contentPane.add(textHiptriangulo);

JLabel lblNewLabel_4_1_3_1 = new
JLabel("Perimetro del triángulo");
    lblNewLabel_4_1_3_1.setFont(new Font("Verdana",
Font.PLAIN, 11));
    lblNewLabel_4_1_3_1.setBounds(897, 474, 323,
14);
contentPane.add(lblNewLabel_4_1_3_1);

textPtriangulo = new JTextField();
textPtriangulo.setColumns(10);
textPtriangulo.setBounds(910, 491, 132, 20);
contentPane.add(textPtriangulo);

JLabel lblNewLabel_5_2_1_1_1 = new JLabel("Area
del triángulo");
    lblNewLabel_5_2_1_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
    lblNewLabel_5_2_1_1_1.setBounds(652, 522, 173,
14);
contentPane.add(lblNewLabel_5_2_1_1_1);

textAtriangulo = new JTextField();
textAtriangulo.setColumns(10);
```

```
textAtriangulo.setBounds(652, 547, 132, 20);
contentPane.add(textAtriangulo);

JLabel lblNewLabel_4_1_3_2 = new JLabel("Ingrese
diámetro menor del rombo");
lblNewLabel_4_1_3_2.setFont(new Font("Verdana",
Font.PLAIN, 11));
lblNewLabel_4_1_3_2.setBounds(212, 617, 323,
14);
contentPane.add(lblNewLabel_4_1_3_2);

textDMenor_rombo = new JTextField();
textDMenor_rombo.setColumns(10);
textDMenor_rombo.setBounds(240, 642, 139, 20);
contentPane.add(textDMenor_rombo);

JLabel lblNewLabel_4_1_3_2_1 = new
JLabel("Ingrese diámetro mayor del rombo");
lblNewLabel_4_1_3_2_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
lblNewLabel_4_1_3_2_1.setBounds(212, 664, 323,
14);
contentPane.add(lblNewLabel_4_1_3_2_1);

textDMayor_rombo = new JTextField();
textDMayor_rombo.setColumns(10);
textDMayor_rombo.setBounds(240, 683, 139, 20);
```

```

        contentPane.add(textDMayor_rombo);

        JButton btnCrombo = new JButton("Calcular");
        btnCrombo.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                try {
                    //CALCULA PROPIEDADES ROMBO

                    double DMenorombo=
Double.parseDouble(textDMenor_rombo.getText()),DMayor=
Double.parseDouble(textDMayor_rombo.getText()),lado=
Double.parseDouble(textLrombo.getText());

                    Rombo mirombo = new
Rombo(lado,DMenorombo,DMayor);

                    textArombo.setText(String.valueOf(mirombo.calcularAr
ea()));

                    textPrombo.setText(String.valueOf(mirombo.calcularPe
rimetro()));

                }catch(NumberFormatException ex) {
                    //EXCEPCION VALORES NO NÚMERICOS

                    JOptionPane.showMessageDialog(null, "Rellene todos
los campos o ingrese un formato valido");

                }
            }
        });
    }
}

```

```

        }
    });
    btnCrombo.setBounds(511, 661, 89, 23);
    contentPane.add(btnCrombo);

    JLabel lblNewLabel_5_2_1_1_1_1 = new
JLabel("Area del rombo");
    lblNewLabel_5_2_1_1_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
    lblNewLabel_5_2_1_1_1_1.setBounds(673, 644, 173,
14);
    contentPane.add(lblNewLabel_5_2_1_1_1_1);

    JLabel lblNewLabel_4_1_3_1_1 = new
JLabel("Perimetro del rombo");
    lblNewLabel_4_1_3_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
    lblNewLabel_4_1_3_1_1.setBounds(910, 644, 323,
14);
    contentPane.add(lblNewLabel_4_1_3_1_1);

    textArombo = new JTextField();
    textArombo.setColumns(10);
    textArombo.setBounds(649, 662, 132, 20);
    contentPane.add(textArombo);

    textPrombo = new JTextField();

```

```
textPrombo.setColumns(10);
textPrombo.setBounds(910, 662, 132, 20);
contentPane.add(textPrombo);

JLabel lblNewLabel_4_1_3_2_2 = new
JLabel("Ingresa base mayor");
    lblNewLabel_4_1_3_2_2.setFont(new
Font("Verdana", Font.PLAIN, 11));
    lblNewLabel_4_1_3_2_2.setBounds(240, 781, 323,
14);
    contentPane.add(lblNewLabel_4_1_3_2_2);

textBMayor_trapezio = new JTextField();
textBMayor_trapezio.setColumns(10);
textBMayor_trapezio.setBounds(240, 807, 139,
20);
    contentPane.add(textBMayor_trapezio);

JLabel lblNewLabel_4_1_3_2_2_1 = new
JLabel("Ingresa base menor");
    lblNewLabel_4_1_3_2_2_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
    lblNewLabel_4_1_3_2_2_1.setBounds(240, 830, 323,
14);
    contentPane.add(lblNewLabel_4_1_3_2_2_1);

textBMenor_trapezio = new JTextField();
```



```

        textBMenor_trapecio.setColumns(10);
        textBMenor_trapecio.setBounds(240, 850, 139,
20);

        contentPane.add(textBMenor_trapecio);


        JLabel lblNewLabel_4_1_3_2_2_1_1 = new
JLabel("Ingrese altura");

        lblNewLabel_4_1_3_2_2_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));

        lblNewLabel_4_1_3_2_2_1_1.setBounds(240, 871,
323, 14);

        contentPane.add(lblNewLabel_4_1_3_2_2_1_1);


        textHtrapecio = new JTextField();
        textHtrapecio.setColumns(10);
        textHtrapecio.setBounds(240, 887, 139, 20);
        contentPane.add(textHtrapecio);


        JButton btnCtrapeccio = new
JButton("Calcular\r\n");

        btnCtrapeccio.addActionListener(new
ActionListener() {

            public void actionPerformed(ActionEvent e)

            {

                try {

                    //CALCULA PROPIEDADES TRAPECIO

                    double BMenor=
Double.parseDouble(textBMenor_trapecio.getText()),BMayor=

```

```

Double.parseDouble(textBMayor_trapecio.getText()), altura=
Double.parseDouble(textHtrapecio.getText());

        Trapecio mitrapecio = new
Trapecio(BMayor,BMenor,altura);

        textAtrapecio.setText(String.valueOf(mitrapecio.calc
ularArea()));

        textPtrapecio.setText(String.valueOf(mitrapecio.calc
ularPerimetro()));

        }catch(NumberFormatException ex) {
                //EXCEPCION VALORES NO NÚMERICOS

        JOptionPane.showMessageDialog(null, "Rellene todos
los campos o ingrese un formato valido");

        }

        }

    });

    btnCtrapecio.setBounds(511, 827, 89, 23);
    contentPane.add(btnCtrapecio);

    JLabel lblNewLabel_5_2_1_1_1_1_1 = new
JLabel("Area del trapecio");

    lblNewLabel_5_2_1_1_1_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));

```

```
        lblNewLabel_5_2_1_1_1_1_1.setBounds(652, 803,  
173, 14);
```

```
        contentPane.add(lblNewLabel_5_2_1_1_1_1_1);
```

```
        textAtrapecio = new JTextField();
```

```
        textAtrapecio.setColumns(10);
```

```
        textAtrapecio.setBounds(652, 828, 132, 20);
```

```
        contentPane.add(textAtrapecio);
```

```
        JLabel lblNewLabel_4_1_3_1_1_1 = new  
JLabel("Perimetro del trapecio");
```

```
        lblNewLabel_4_1_3_1_1_1.setFont(new  
Font("Verdana", Font.PLAIN, 11));
```

```
        lblNewLabel_4_1_3_1_1_1.setBounds(910, 803, 323,  
14);
```

```
        contentPane.add(lblNewLabel_4_1_3_1_1_1);
```

```
        textPtrapecio = new JTextField();
```

```
        textPtrapecio.setColumns(10);
```

```
        textPtrapecio.setBounds(910, 828, 132, 20);
```

```
        contentPane.add(textPtrapecio);
```

```
        JLabel lblNewLabel_5_2_1_1_1_2 = new  
JLabel("Tipo de triángulo");
```

```
        lblNewLabel_5_2_1_1_1_2.setFont(new  
Font("Verdana", Font.PLAIN, 11));
```

```

14);
        lblNewLabel_5_2_1_1_1_2.setBounds(903, 522, 173,

        contentPane.add(lblNewLabel_5_2_1_1_1_2);

        textTipotriangulo = new JTextField();
        textTipotriangulo.setColumns(10);
        textTipotriangulo.setBounds(910, 547, 200, 20);
        contentPane.add(textTipotriangulo);

        JLabel lblNewLabel_4_1_3_2_1_1 = new
JLabel("Ingrese lado del rombo");
        lblNewLabel_4_1_3_2_1_1.setFont(new
Font("Verdana", Font.PLAIN, 11));
        lblNewLabel_4_1_3_2_1_1.setBounds(229, 707, 323,
14);

        contentPane.add(lblNewLabel_4_1_3_2_1_1);

        textLrombo = new JTextField();
        textLrombo.setColumns(10);
        textLrombo.setBounds(240, 728, 139, 20);
        contentPane.add(textLrombo);

        JButton btnBorrarC = new JButton("Borrar");
        btnBorrarC.addActionListener(new
ActionListener() {
            public void actionPerformed(ActionEvent e)
{

```

```

        textRcirculo.setText("");
        textAcirculo.setText("");
        textPcirculo.setText("");
    }
});
btnBorrarC.setBounds(1131, 116, 89, 23);
contentPane.add(btnBorrarC);

JButton btnBorrarR = new JButton("Borrar");
btnBorrarR.addActionListener(new
ActionListener() {
    public void actionPerformed(ActionEvent e)
    {

        textBrectangulo.setText("");
        textHrectangulo.setText("");
        textArectangulo.setText("");
        textPrectangulo.setText("");
    }
});
btnBorrarR.setBounds(1131, 233, 89, 23);
contentPane.add(btnBorrarR);

JButton btnCuadrado = new JButton("Borrar");
btnCuadrado.addActionListener(new
ActionListener() {

```

```

        public void actionPerformed(ActionEvent e)
    {

        textLcuadrado.setText("");
        textAcuadrado.setText("");
        textPcuadrado.setText("");

    }

});

btnCuadrado.setBounds(1131, 369, 89, 23);
contentPane.add(btnCuadrado);

JButton btnBorrarrec = new JButton("Borrar");
btnBorrarrec.addActionListener(new
ActionListener() {
    public void actionPerformed(ActionEvent e)
    {

        textTipotriangulo.setText("");
        textBtriangulo.setText("");
        textHtriangulo.setText("");
        textHiptriangulo.setText("");
        textAtriangulo.setText("");
        textPtriangulo.setText("");

    }

});

```

```

        btnBorrarrec.setBounds(1131, 499, 89, 23);
        contentPane.add(btnBorrarrec);

        JButton btnBorrartri = new JButton("Borrar");
        btnBorrartri.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
        {
                textDMenor_rombo.setText("");
                textDMayor_rombo.setText("");
                textLrombo.setText("");
                textArombo.setText("");
                textPrombo.setText("");
            }
        });
        btnBorrartri.setBounds(1131, 661, 89, 23);
        contentPane.add(btnBorrartri);

        JButton btnBorrartra = new JButton("Borrar");
        btnBorrartra.addActionListener(new
        ActionListener() {
            public void actionPerformed(ActionEvent e)
        {
                textBMenor_trapecio.setText("");
                textBMayor_trapecio.setText("");
                textHtrapecio.setText("");
            }
        });

```

```
        textAtrapecio.setText("");
        textPtrapecio.setText("");

    }

});

btnBorrartra.setBounds(1131, 821, 89, 23);
contentPane.add(btnBorrartra);

}

}
```