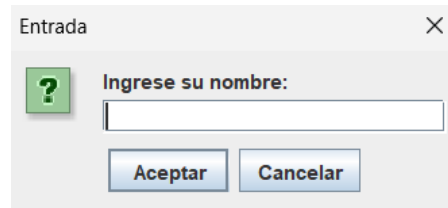


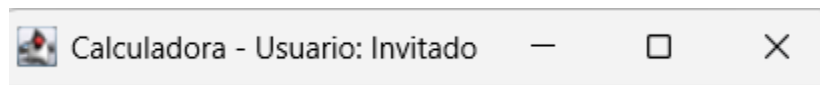
Funcionamiento:

- Al iniciar la aplicación se mostrará una ventana en la cual le pedirá al usuario que coloque su nombre para que el sistema registre quien es quien ejecuta las operaciones.

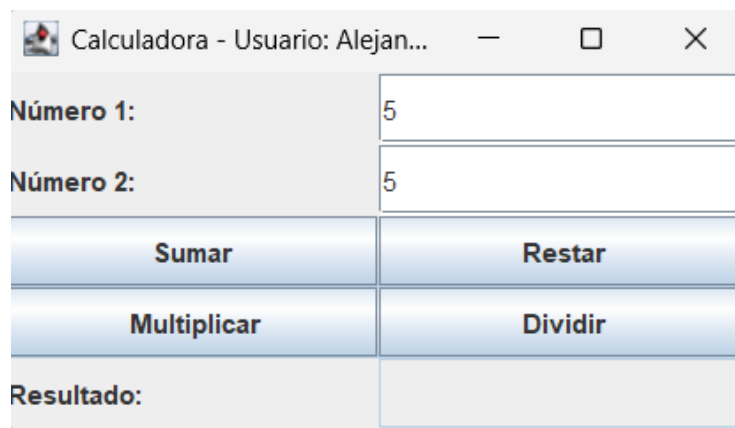


A screenshot of a Windows-style dialog box titled "Entrada" with a close button (X) in the top right corner. Inside the dialog, there is a green square icon with a white question mark. To the right of the icon is the text "Ingrese su nombre:" followed by a text input field. Below the input field are two buttons: "Aceptar" and "Cancelar".

- Si en caso el usuario no coloca su nombre y le da aceptar el sistema seguirá en modo invitado y en la ventana se mostrará

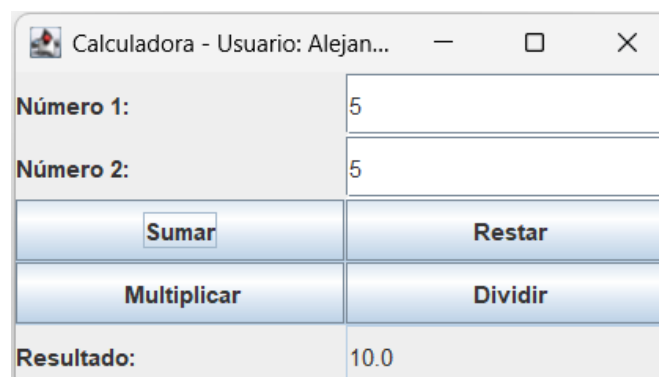


- Después se mostrará una ventana nueva en donde se deberán colocar 2 números y con ellos el usuario deberá colocar que operación desea realizar con esos números.



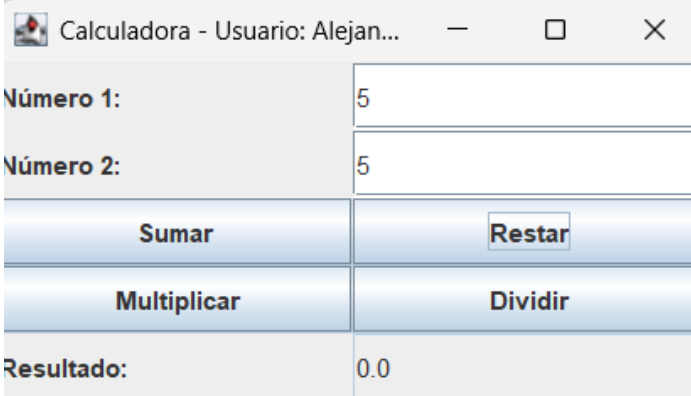
A screenshot of the main interface of a calculator application window titled "Calculadora - Usuario: Alejan...". The window has standard minimize, maximize, and close buttons. The interface is divided into two columns. The left column contains labels: "Número 1:", "Número 2:", and "Resultado:". The right column contains input fields and buttons. The first input field contains the number "5". The second input field contains the number "5". Below the input fields are four buttons arranged in a 2x2 grid: "Sumar", "Restar", "Multiplicar", and "Dividir". The "Resultado:" label is at the bottom left, and its corresponding input field is empty.

- Suma:



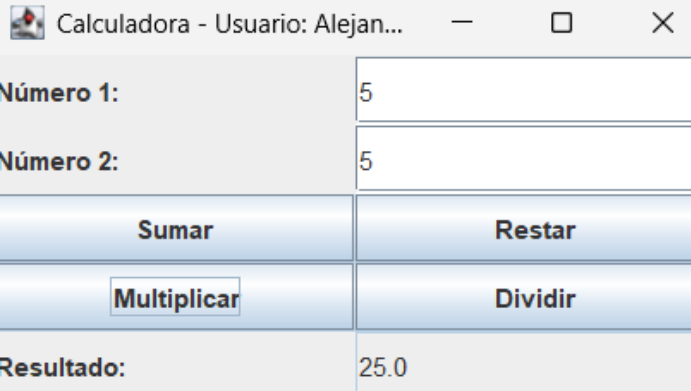
A screenshot of the calculator application window after a sum operation. The window title is "Calculadora - Usuario: Alejan...". The interface is the same as the previous screenshot, but the "Resultado:" input field now displays the value "10.0". The "Sumar" button is highlighted with a blue border.

- Resta:



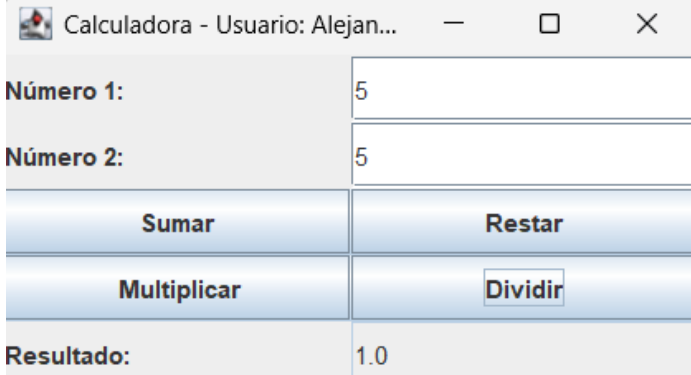
Número 1:	5
Número 2:	5
Sumar	Restar
Multiplicar	Dividir
Resultado:	0.0

- Multiplicación



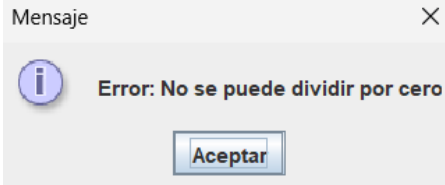
Número 1:	5
Número 2:	5
Sumar	Restar
Multiplicar	Dividir
Resultado:	25.0

- División:




Número 1:	5
Número 2:	5
Sumar	Restar
Multiplicar	Dividir
Resultado:	1.0

- Si dado caso el usuario decide hacer división con 0 se le mostrará lo siguiente:



Mensaje

 Error: No se puede dividir por cero

Aceptar

- Ahí se mostrará con más detalle quien realiza las acciones que fue el nombre que pide al iniciar.

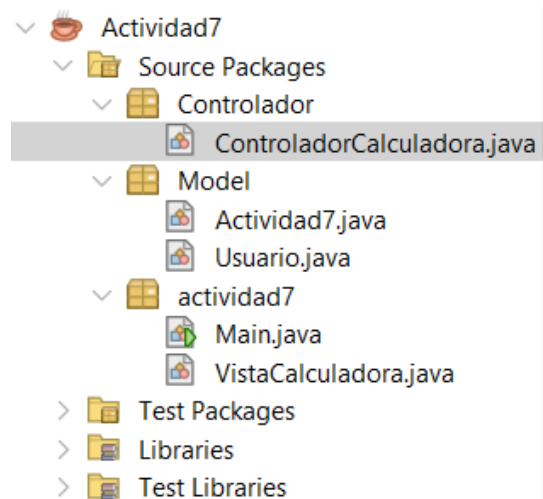
```

Usuario Alejandro realizo: sumar -> Resultado: 10.0
Usuario Alejandro realizo: restar -> Resultado: 0.0
Usuario Alejandro realizo: multiplicar -> Resultado: 25.0
Usuario Alejandro realizo: dividir -> Resultado: 1.0
Usuario Alejandro realizo: sumar -> Resultado: 5.0
Usuario Alejandro realizo: restar -> Resultado: 5.0
Usuario Alejandro realizo: multiplicar -> Resultado: 0.0

```

Estructura del Proyecto:

- La estructura general es la siguiente:



- En el primer paquete Controlador se encuentra ControladorCalculadora, el cual conecta los botones de la vista con los métodos del modelo que realizan operaciones matemáticas. Además, gestiona errores y muestra los resultados al usuario. También registra en consola qué usuario hizo qué operación.

```

package Controlador;
import Model.Actividad7;
import Model.Usuario;
import actividad7.VistaCalculadora;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class ControladorCalculadora {

    private Actividad7 modelo;
    private Usuario usuario;
    private VistaCalculadora vista;

    public ControladorCalculadora(Actividad7 modelo, Usuario usuario, VistaCalculadora vista) {
        this.modelo = modelo;
        this.usuario = usuario;
        this.vista = vista;

        this.vista.botonSumar.addActionListener(new OperacionListener("sumar"));
        this.vista.botonRestar.addActionListener(new OperacionListener("restar"));
        this.vista.botonMultiplicar.addActionListener(new OperacionListener("multiplicar"));
        this.vista.botonDividir.addActionListener(new OperacionListener("dividir"));
    }

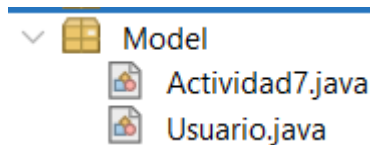
    private class OperacionListener implements ActionListener {
        private String operacion;

        public OperacionListener(String operacion) {
            this.operacion = operacion;
        }

        @Override
        public void actionPerformed(ActionEvent e) {
            try {
                double num1 = Double.parseDouble(vista.campoNumero1.getText());
                double num2 = Double.parseDouble(vista.campoNumero2.getText());
                double resultado = 0;

```

- En el paquete model se encuentran dos clases las cuales son:



- La clase Actividad7 está en el paquete Model y representa la lógica de negocio del sistema, es decir, las operaciones matemáticas reales que puede ejecutar la calculadora.

```

package Model;
public class Actividad7 {

    public double sumar(double a, double b) {
        return a + b;
    }

    public double restar(double a, double b) {
        return a - b;
    }

    public double multiplicar(double a, double b) {
        return a * b;
    }

    public double dividir(double a, double b) {
        if (b == 0) {
            throw new ArithmeticException("No se puede dividir por cero");
        }
        return a / b;
    }
}

```

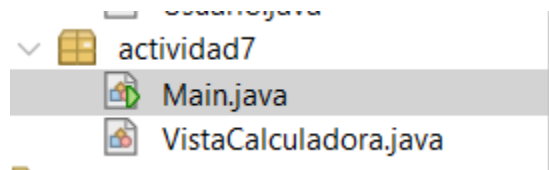
- La clase usuario identifica al usuario actual y lo muestra en la ventana general.

```
package Model;
public class Usuario {
    private String nombre;

    public Usuario(String nombre) {
        this.nombre = nombre;
    }

    public String getNombre() {
        return nombre;
    }
}
```

- El paquete actividad7 contiene a:



- La clase Main es de las mas importantes ya que es esta en donde empieza prácticamente todo, en ella se inicia el programa y es en donde le pide al usuario que coloque su nombre para empezar a realizar la calculadora.

```
package actividad7;
import Model.Usuario;
import Model.Actividad7;
import Controlador.ControladorCalculadora;
import javax.swing.*;

public class Main {
    public static void main(String[] args) {
        String nombreUsuario = JOptionPane.showInputDialog("Ingrese su nombre:");
        if (nombreUsuario == null || nombreUsuario.trim().isEmpty()) {
            nombreUsuario = "Invitado";
        }

        Actividad7 modelo = new Actividad7();
        Usuario usuario = new Usuario(nombreUsuario);
        VistaCalculadora vista = new VistaCalculadora(nombreUsuario);

        new ControladorCalculadora(modelo, usuario, vista);
    }
}
```

- La clase VistaCalculadora es la interfaz gráfica de la aplicación de calculadora. Su función principal es:

Mostrar al usuario una ventana donde puede ingresar dos números, elegir una operación (sumar, restar, multiplicar o dividir), y ver el resultado.

```
package actividad7;
import javax.swing.*;
import java.awt.*;

public class VistaCalculadora extends JFrame {
    public JTextField campoNumero1 = new JTextField(10);
    public JTextField campoNumero2 = new JTextField(10);
    public JTextField campoResultado = new JTextField(15);

    public JButton botonSumar = new JButton("Sumar");
    public JButton botonRestar = new JButton("Restar");
    public JButton botonMultiplicar = new JButton("Multiplicar");
    public JButton botonDividir = new JButton("Dividir");

    public VistaCalculadora(String nombreUsuario) {
        setTitle("Calculadora - Usuario: " + nombreUsuario);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(350, 200);
        setLayout(new GridLayout(5, 2));

        add(new JLabel("Número 1:"));
        add(campoNumero1);

        add(new JLabel("Número 2:"));
        add(campoNumero2);

        add(botonSumar);
        add(botonRestar);

        add(botonMultiplicar);
        add(botonDividir);

        add(new JLabel("Resultado:"));
        campoResultado.setEditable(false);
        add(campoResultado);

        setVisible(true);
    }

    public void mostrarMensaje(String mensaje) {
        JOptionPane.showMessageDialog(this, mensaje);
    }
}
```

