Seguimiento 6

Presentado por:  
Nicolás Pérez Cuasquer

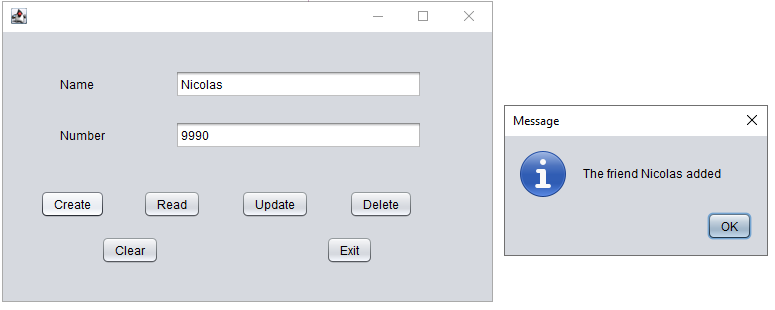
cc. 1.214.746.554

Presentado a:  
Walter Hugo Arboleda Mazo

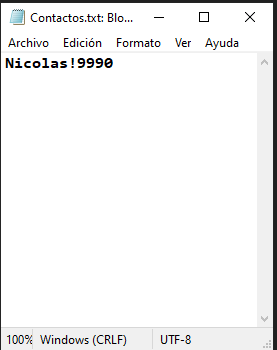
Universidad Nacional de Colombia

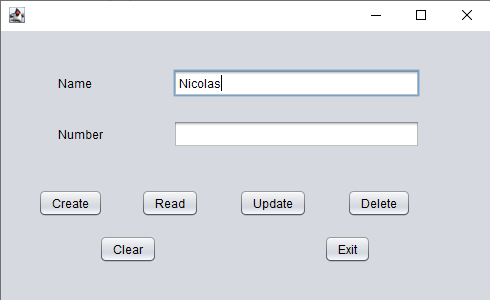
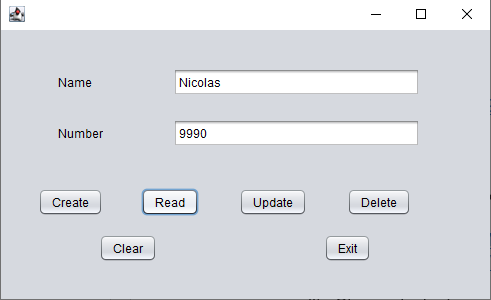
Programación orientada a objetos

2023-2

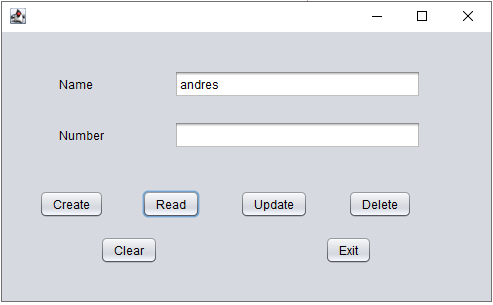
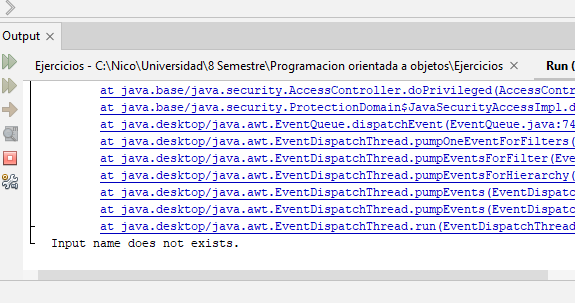
1. **Imagen de la interfaz de usuario:**

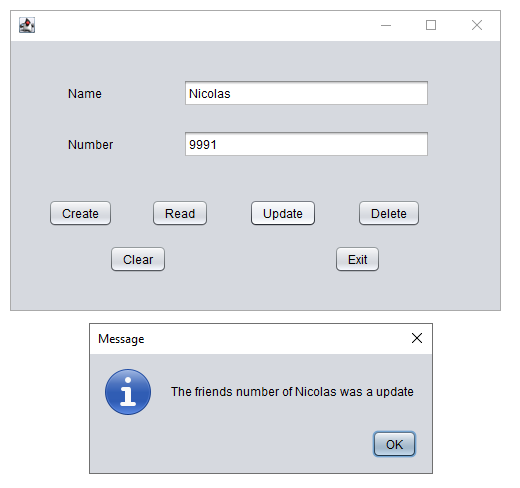
* **Botón “Create”:**

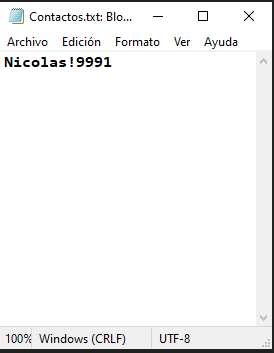
****

* **Botón “Read”:**

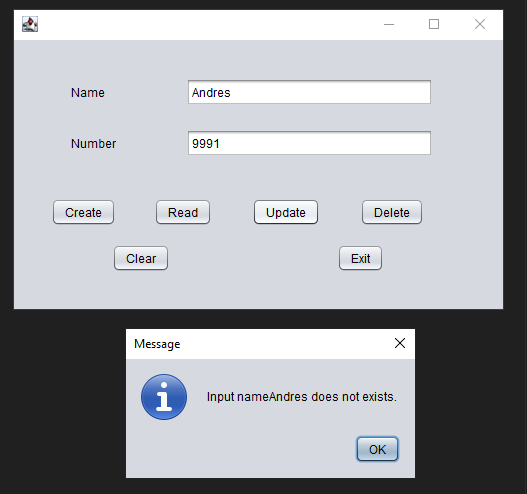
**En caso de no encontrar:**

****

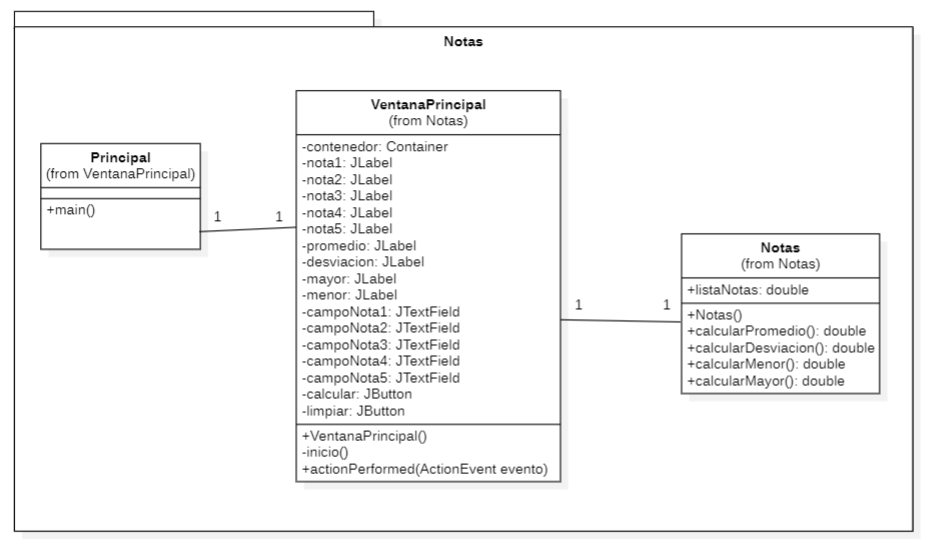
* **Botón “Update”:**

****

**En caso de no encontrar:**

****

* **Diagrama de clases**:
* **Diagrama casos de uso:**
* **ENLACE GENERAL DEL PROYECTO EN GITHUB**:

****<https://github.com/Mrnicolas1999/POOEjercicios/tree/main/Seguimiento%205/POOEP1/src/main/java/Notas>

* **ENLACE DE CADA UNA DE LAS CLASES:**
  + 1. Clase Notas: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP1/src/main/java/Notas/Notas.java>
    2. Clase principal: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP1/src/main/java/Notas/Principal.java>
    3. Clase Ventana Principal:

<https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP1/src/main/java/Notas/VentanaPrincipal.java>

* **CÓDIGO POR CLASES: Clase notas:**

package Notas;

public class Notas {

double[] listaNotas; /\* Atributo que identifica un array de notas de tipo double \*/

public Notas() {

listaNotas = new double[5]; // Crea un array de 5 notas

}

public double calcularPromedio() {

double suma = 0;

for(int i=0; i < listaNotas.length; i++) { // Se recorre el array

suma = suma + listaNotas[i]; // Suma las notas del array

}

sobre el total de notas \*/

return (suma / listaNotas.length);

}

public double calcularDesviacion() {

double prom = calcularPromedio(); /\* Invoca el método para calcular el promedio \*/

double suma = 0;

for(int i=0; i < listaNotas.length; i++) { // Aplica fórmula para la sumatoria de elementos

suma += Math.pow(listaNotas[i] - prom, 2 );

}

return Math.sqrt(suma/listaNotas.length ); /\* Retorna el cálculo final de la desviación \*/

}

public double calcularMenor() {

double menor = listaNotas[0]; /\* Define una variable como la nota menor \*/

for(int i=0; i < listaNotas.length; i++) { // Se recorre el array

if (listaNotas[i] < menor) {

/\* Si un elemento del array es menor que el menor actual,se actualiza su valor \*/

menor = listaNotas[i];

}

}

return menor;

}

public double calcularMayor() {

double mayor = listaNotas[0]; /\* Define una variable como la nota mayor \*/

for(int i=0; i < listaNotas.length; i++) { // Se recorre el array

if (listaNotas[i] > mayor) {

/\* Si un elemento del array es mayor que el mayor actual, se actualiza su valor \*/

mayor = listaNotas[i];

}

}

return mayor;

}

}

**Clase principal:**

package Notas;

public class Principal {

public static void main(String[] args) {

VentanaPrincipal miVentanaPrincipal; /\* Define la ventana principal \*/

miVentanaPrincipal= new VentanaPrincipal(); /\* Crea la ventana principal \*/

miVentanaPrincipal.setVisible(true); /\* Establece la ventana como visible \*/

}

}

**Clase ventana principal:**

package Notas;

/\*\*

\*

\* @author Nicolas

\*/

public class VentanaPrincipal extends javax.swing.JFrame {

public VentanaPrincipal() {

initComponents();

setTitle("Notas"); // Establece el título de la ventana

setSize(280,380); // Establece el tamaño de la ventana

setLocationRelativeTo(null);

setDefaultCloseOperation(VentanaPrincipal.EXIT\_ON\_CLOSE);

setResizable(false);

}

private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnCalcularActionPerformed

Notas notas = new Notas();

notas.listaNotas[0] = Double.parseDouble(campoNota1.getText());

notas.listaNotas[1] = Double.parseDouble(campoNota2.getText());

notas.listaNotas[2] = Double.parseDouble(campoNota3.getText());

notas.listaNotas[3] = Double.parseDouble(campoNota4.getText());

notas.listaNotas[4] = Double.parseDouble(campoNota5.getText());

notas.calcularPromedio(); // Se calcula el promedio

notas.calcularDesviacion(); // Se calcula la desviación

promedio.setText("Promedio = " + String.valueOf(String.format("%.2f",notas.calcularPromedio())));

double desv = notas.calcularDesviacion();

// Se muestra la desviación formateada

desviacion.setText("Desviación estándar = " + String.format("%.2f", desv));

mayor.setText("Valor mayor = " + String.valueOf(notas.calcularMayor()));

menor.setText("Valor menor = " + String.valueOf(notas.calcularMenor()));

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

campoNota1.setText("");

campoNota2.setText("");

campoNota3.setText("");

campoNota4.setText("");

campoNota5.setText("");

promedio.setText("");

desviacion.setText("");

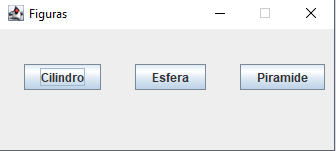
mayor.setText("");

menor.setText("");

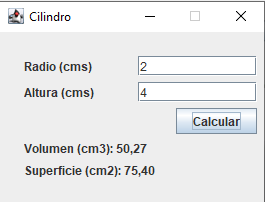
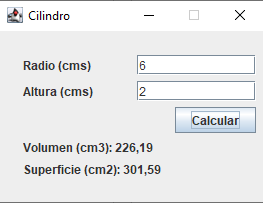
}

1. **EJERCICIO 8.3:**

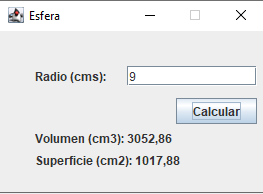
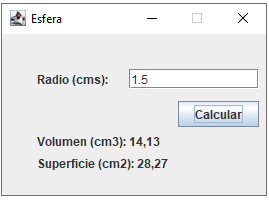
* **Imagen de la interfaz de usuario:**

****

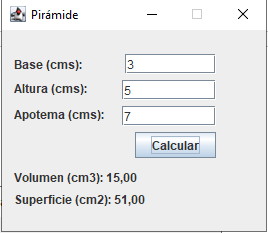
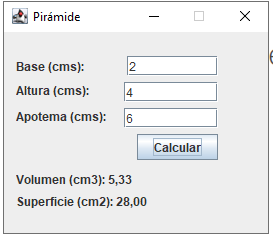
* **Imagen Interfaz de usuario cilindro:**



* **Imagen Interfaz de usuario esfera:**



* **Imagen Interfaz de usuario pirámide:**



* **ENLACE GENERAL DEL PROYECTO EN GITHUB:**

<https://github.com/Mrnicolas1999/POOEjercicios/tree/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras>

* **ENLACES POR CLASES:**
  + 1. Clase cilindro: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/Cilindro.java>
    2. Clase esfera: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/Esfera.java>
    3. Clase FiguraGeometrica: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/FiguraGeometrica.java>
    4. Clase pirámide: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/Piramide.java>
    5. Clase Principal: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/Principal.java>
    6. Clase VentanaCilindro: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/VentanaCilindro.java>
    7. Clase VentanaEsfera: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/VentanaEsfera.java>
    8. Clase VentanaPiramide: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/VentanaPiramide.java>
    9. Clase VentanaPrincipal: <https://github.com/Mrnicolas1999/POOEjercicios/blob/main/Seguimiento%205/POOEP2/POOEP2/src/main/java/Figuras/VentanaPrincipal.java>

**CÓDIGO POR CLASES: Clase Cilindro:**

package Figuras;

public class Cilindro extends FiguraGeometrica {

private double radio; // Atributo que establece el radio de un cilindro

private double altura; // Atributo que establece la altura de un cilindro

public Cilindro(double radio, double altura) {

this.radio = radio;

this.altura = altura;

this.setVolumen(calcularVolumen()); /\* Calcula el volumen y establece su atributo \*/

this.setSuperficie(calcularSuperficie()); /\* Calcula la superficie y establece su atributo \*/

}

public double calcularVolumen() {

double volumen = Math.PI \* altura \* Math.pow(radio, 2.0);

return volumen;

}

public double calcularSuperficie() {

double áreaLadoA = 2.0 \* Math.PI \* radio \* altura;

double áreaLadoB = 2.0 \* Math.PI \* Math.pow(radio, 2.0);

return áreaLadoA + áreaLadoB;

}

}

**Clase Esfera:**

package Figuras;

public class Esfera extends FiguraGeometrica {

private double radio; // Atributo que identifica el radio de una esfera/

public Esfera(double radio) {

this.radio = radio;

this.setVolumen(calcularVolumen()); /\* Calcula el volumen y establece su atributo \*/

this.setSuperficie(calcularSuperficie()); /\* Calcula la superficie y establece su atributo \*/

}

public double calcularVolumen() {

double volumen = 1.333 \* Math.PI \* Math.pow(this.radio, 3.0);

return volumen;

}

public double calcularSuperficie() {

double superficie = 4.0 \* Math.PI \* Math.pow(this.radio, 2.0);

return superficie;

}

}

**Clase pirámide:**

package Figuras;

public class Piramide extends FiguraGeometrica {

private double base; /\* Atributo que identifica la base de una pirámide \*/

private double altura; /\* Atributo que identifica la altura de una pirámide \*/

private double apotema; /\* Atributo que identifica el apotema de una pirámide \*/

public Piramide(double base, double altura, double apotema) {

this.base = base;

this.altura = altura;

this.apotema = apotema;

this.setVolumen(calcularVolumen()); /\* Calcula el volumen y establece su atributo \*/

this.setSuperficie(calcularSuperficie()); /\* Calcula la superficie y establece su atributo \*/

}

public double calcularVolumen() {

double volumen = (Math.pow(base, 2.0) \* altura) / 3.0;

return volumen;

}

public double calcularSuperficie() {

double áreaBase = Math.pow(base, 2.0);

double áreaLado = 2.0 \* base \* apotema;

return áreaBase + áreaLado;

}

}

**Clase Figura Geométrica:**

package Figuras;

public class FiguraGeometrica {

private double volumen; /\* Atributo que identifica el volumen deuna figura geométrica \*/

private double superficie; /\* Atributo que identifica la superficie de una figura geométrica \*/

public void setVolumen(double volumen) {

this.volumen = volumen;

}

public void setSuperficie(double superficie) {

this.superficie = superficie;

}

public double getVolumen() {

return this.volumen;

}

public double getSuperficie() {

return this.superficie;

}

}

**Clase Principal:**

package Figuras;

public class Principal {

public static void main(String[] args) {

VentanaPrincipal miVentanaPrincipal; /\* Define la ventana principal \*/

miVentanaPrincipal= new VentanaPrincipal(); /\* Crea la ventana principal \*/

miVentanaPrincipal.setVisible(true);

miVentanaPrincipal.setResizable(false);

}

}

**Clase VentanaCilindro:**

package Figuras;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class VentanaCilindro extends javax.swing.JFrame {

public VentanaCilindro() {

initComponents();

setTitle("Cilindro");// Establece el título de la ventana\*/

setSize(280,210); // Establece el tamaño de la ventana\*/

setLocationRelativeTo(null); /\* La ventana se posiciona en el centro de la pantalla \*/

setResizable(false); /\* Establece que el botón de cerrar permitirá salir de la aplicación \*/

}

private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnCalcularActionPerformed

boolean error = false; /\* Se inicializa variable para determinar si ocurre un error \*/

double radio = 0;

double altura = 0;

try {

radio = Double.parseDouble(campoRadio.getText());

altura = Double.parseDouble(campoAltura.getText());

Cilindro cilindro = new Cilindro(radio, altura);

volumen.setText("Volumen (cm3): " + String.format("%.2f",cilindro.calcularVolumen()));

superficie.setText("Superficie (cm2): " + String.format("%.2f", cilindro.calcularSuperficie()));

} catch (Exception e){

error = true; // Si ocurre una excepción

} finally {

if(error) { /\* Si ocurre una excepción, se muestra un mensaje de error \*/

JOptionPane.showMessageDialog(null,"Campo nulo o error en formato de numero","Error", JOptionPane.ERROR\_MESSAGE);

}

}

}

**Clase VentanaEsfera:**

package Figuras;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class VentanaEsfera extends javax.swing.JFrame {

public VentanaEsfera() {

initComponents();

setTitle("Esfera"); // Establece el título de la ventana

setSize(280,200); // Establece el tamaño de la ventana

setLocationRelativeTo(null); /\* La ventana se posiciona en el centro de la pantalla \*/

setResizable(false); /\* Establece que el botón de cerrar permitirá salir de la aplicación \*/

}private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {

try {

double radio = Double.parseDouble(campoRadio.getText());

Esfera esfera = new Esfera(radio); /\* Se crea un objeto Esfera \*/

volumen.setText("Volumen (cm3): " + String.format("%.2f", esfera.calcularVolumen()));

superficie.setText("Superficie (cm2): " + String.format("%.2f",esfera.calcularSuperficie()));

} catch (Exception e) {

error = true; // Si ocurre una excepción

} finally {

if(error) { /\* Si ocurre una excepción, se muestra un mensaje de error \*/

JOptionPane.showMessageDialog(null,"Campo nulo o error en formato de número","Error", JOptionPane.ERROR\_MESSAGE);

}

}

}

}

**Clase VentanaPiramide:**

package Figuras;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class VentanaPiramide extends javax.swing.JFrame {

public VentanaPiramide() {

initComponents();

setTitle("Pirámide"); // Establece el título de la ventana

setSize(280,240); // Establece el tamaño de la ventana

setLocationRelativeTo(null); /\* La ventana se posiciona en el centro de la pantalla \*/

setResizable(false); /\* Establece que el botón de cerrar permitirá salir de la aplicación \*/}

private void btnCalcularActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnCalcularActionPerformed

Piramide pirámide;

boolean error = false;

double base = 0;

double altura = 0;

double apotema = 0;

try {

base = Double.parseDouble(campoBase.getText());

altura = Double.parseDouble(campoAltura.getText());

apotema = Double.parseDouble(campoApotema.getText());

pirámide = new Piramide(base, altura, apotema);

volumen.setText("Volumen (cm3): " + String.format("%.2f",pirámide.calcularVolumen()));

superficie.setText("Superficie (cm2): " + String.format("%.2f",pirámide.calcularSuperficie()));

} catch (Exception e) {

error = true; // Si ocurre una excepción

} finally {

if (error) { /\* Si ocurre una excepción, se muestra un mensaje de error \*/

JOptionPane.showMessageDialog(null, "Campo nulo o error en formato de número", "Error",JOptionPane.ERROR\_MESSAGE);

}

}

}

**Clase VentanaPrincipal:**

package Figuras;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class VentanaPrincipal extends javax.swing.JFrame {

public VentanaPrincipal() {

initComponents();

setTitle("Figuras"); // Establece el título de la ventana

setSize(350,160); // Establece el tamaño de la ventana

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}private void btnCilindroActionPerformed(java.awt.event.ActionEvent evt) {

VentanaCilindro cilindro = new VentanaCilindro(); /\* Crea la ventana del cilindro \*/

cilindro.setVisible(true); /\* Establece que se visualice la ventana del cilindro \*/

}private void btnEsferaActionPerformed(java.awt.event.ActionEvent evt) {

VentanaEsfera esfera = new VentanaEsfera(); /\* Crea la ventana de la esfera \*/

esfera.setVisible(true); /\* Establece que se visualice la ventana de la esfera \*/

} private void btnPiramideActionPerformed(java.awt.event.ActionEvent evt) {

VentanaPiramide pirámide = new VentanaPiramide(); /\* Crea la ventana de la pirámide \*/

pirámide.setVisible(true); /\* Establece que se visualice la ventana de la pirámide \*/

}