

**INSTITUTO SUPERIOR TECNOLOGICO PRIVADO CIBERTEC**

**CARRERAS PROFESIONALES**

**INTRODUCCION A LA ALGORITMIA**

**SISTEMA DE VENTA DE CERÁMICOS**

**PRIMER CICLO**

**SECCIÓN T1BL**

**SEMESTRE 2022 – 01**

**Coordinador:**

**Johnny Nestor Jair Ramos Paucar**

**Integrantes:**

**Juan Sebastian Guzman Ramos**

**Mijael Jefferson Esquivel Almanza**

**Laysson Polo De La Cruz**

**CARRERAS PROFESIONALES**

**CIBERTEC**

**LIMA, 2023**

#### Índice

|  |  |
| --- | --- |
| Introducción |  |
| Justificación |  |
| Objetivos |  |
| Definición |  |
| Entregables |  |
| Conclusiones |  |
| Recomendaciones |  |
| Bibliografía |  |

#### Introducción

Este proyecto se centra en el desarrollo de software destinado a la venta de cerámicas. Su objetivo principal es permitir a los clientes personalizar e interactuar con las ventas que desean realizar. La intención es mejorar la calidad del servicio ofrecido hacia el cliente y fortalecer la percepción que este último tiene de la empresa.

#### Justificación

Los mayores aportes que contribuyen al producto es reducir y facilitar el entorno a los encargados de la venta de productos u atención al cliente. Los beneficiaros directos: La mayor parte en tomar posesión del sector son los clientes, ya que estos optan por un servicio mas eficiente y de mejor índole que te va a conceder la empresa, los beneficiarios secundarios serian los desarrolladores del software, otros beneficiaros serian los trabajadores de la empresa. Como modelo tenemos al personal encargado del servicio al cliente de la empresa, ya que se le seria más fácil y eficiente el servicio que ofrece al cliente. Los beneficiaros indirectos: Serian las alianzas y socios que tenga la empresa(cerámicas) correspondiente serian mas eficaces con el producto.

La justificación principal radica en su capacidad para disminuir la carga de trabajo de los empleados responsables de la venta de productos y la atención al cliente. Los principales beneficiarios directos son los clientes, ya que tienen acceso a un nuevo servicio informático de alta calidad proporcionado por la empresa.

Los beneficiarios directos de este proyecto son:

• **Clientes**: Tendrán acceso a un servicio más eficiente y de mejor calidad proporcionado por la empresa.

• **Programadores**: Experimentarán beneficios directos al crear y mantener la herramienta informática diseñada para facilitar el entorno de ventas y atención al cliente.

• **Trabajadores de la Empresa**: Específicamente, el personal encargado del servicio al cliente, que verá facilitada su labor y aumentada su eficiencia.

Los beneficiarios indirectos de este proyecto:

• **Alianzas y socios de la Empresa**: Experimentarán una mayor eficacia en la colaboración con la empresa, ya que el producto contribuirá a optimizar los procesos relacionados con la venta de productos y la atención al cliente.

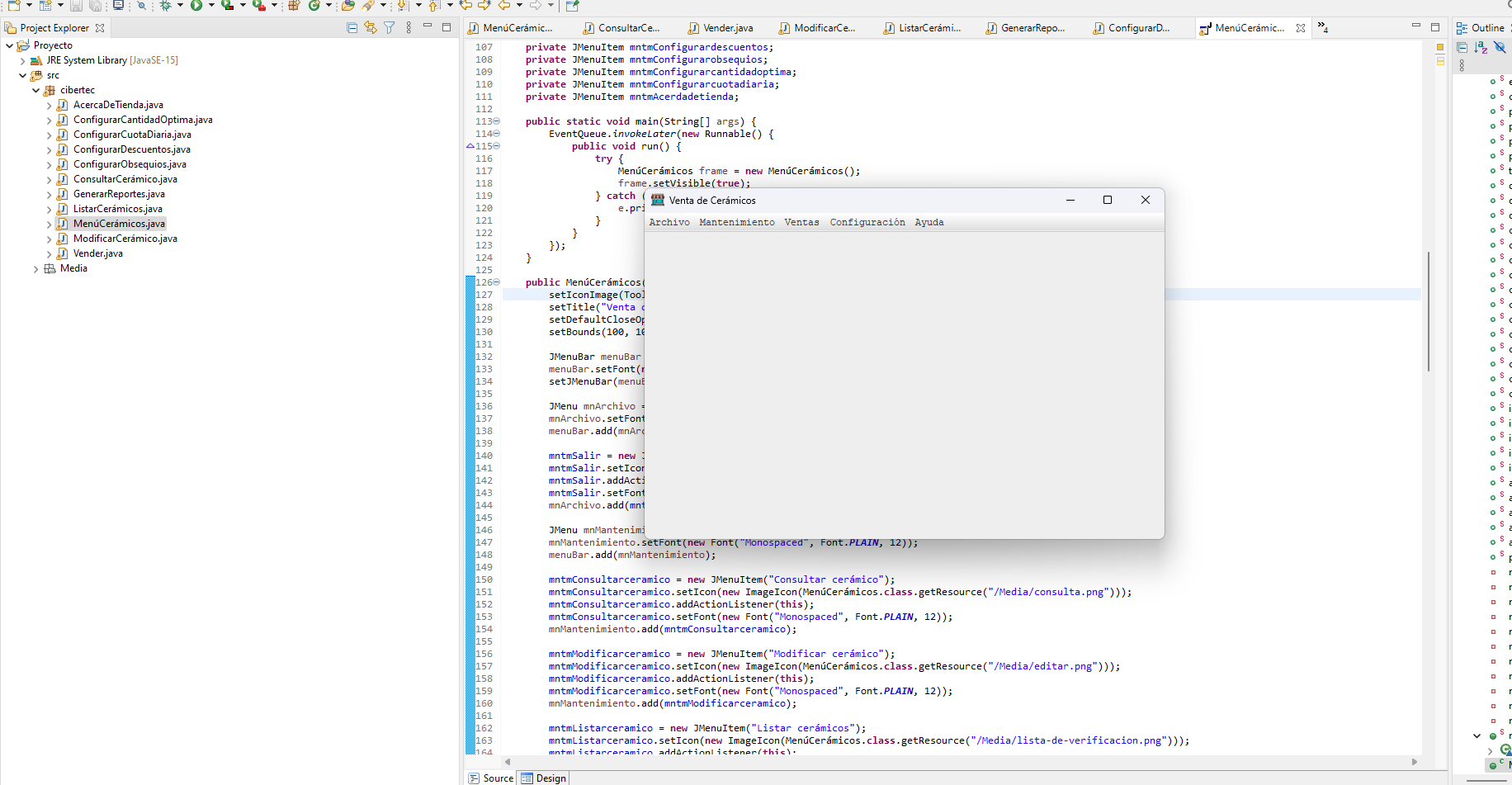
• **Compradores de los productos**: Tendrán una mayor eficiencia en la rápidez de la venta.

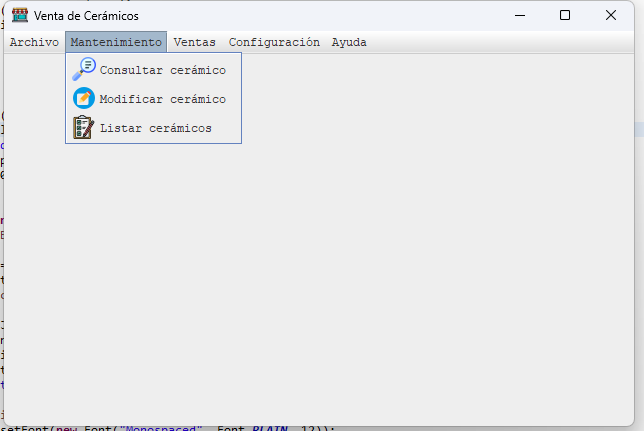
#### Objetivos

•Los objetivos específicos del software de venta de cerámicas son precisas ya que permiten a los clientes interactuar con el sistema y con los productos que ofrecemos.   
•Estos objetivos son medibles, ya que se pueden evaluar y seguir para determinar si estamos cumpliendo con ellos o no.  
•Son alcanzables porque cada persona tiene metas claras que son factibles de lograr y están motivadas para alcanzar.  
•Además, estos objetivos son relevantes ya que se ajustan a nuestras capacidades y recursos disponibles.  
•En cuanto al factor tiempo, es crucial ya que tenemos plazos limitados para cumplir con los objetivos establecidos. Damos prioridad a las tareas con plazos más cercanos y dejamos para más tarde las menos urgentes.

#### Definición

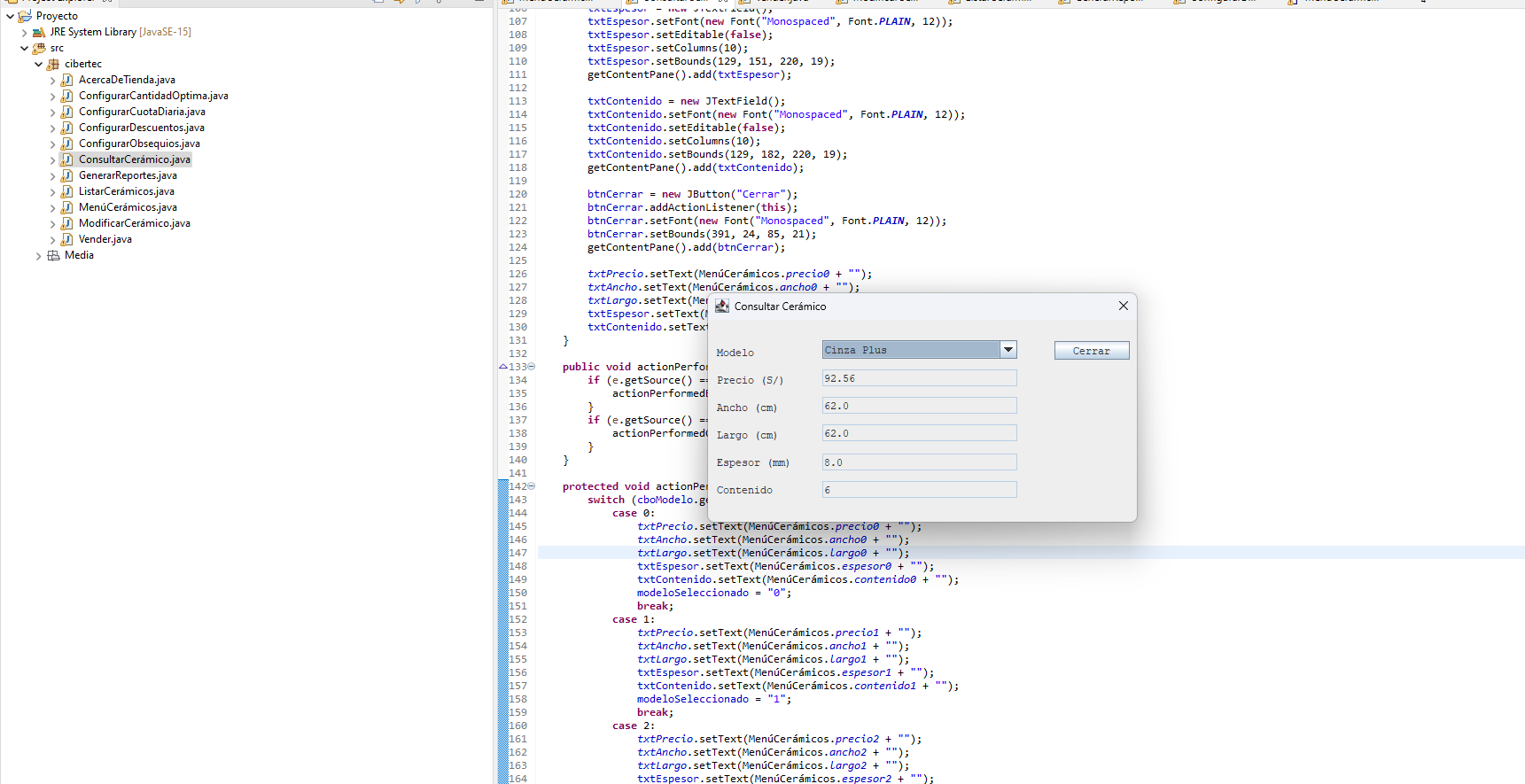
**Menú Principal:** Actúa como una especie de punto central que conecta todas las cajas de diálogo o ventanas de la aplicación en un solo JFrame. Esto significa que desde el Menú Principal, los usuarios pueden acceder a diferentes funciones o características de la aplicación a través de ventanas emergentes o diálogos, a la vez que se le puede agregar una imagen personalizada a gusto del cliente.

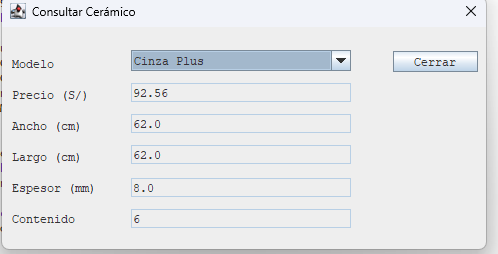




<https://i.imgur.com/dZHl95g.jpg>

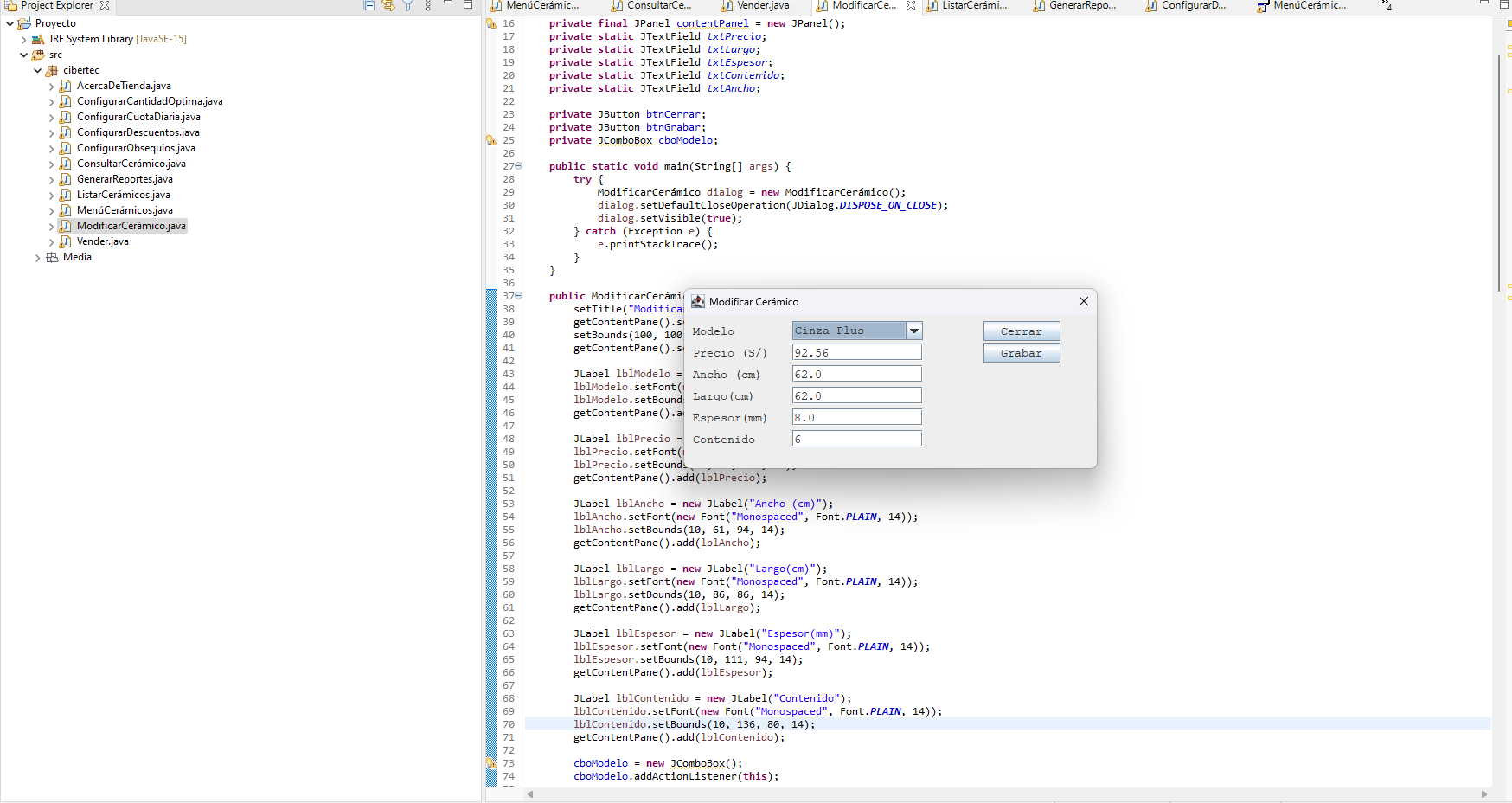
**Consultar Cerámicos:** La función que tiene esta caja de diálogo es otorgar información de los productos al cliente, y el poder consultar los atributos que tiene cada producto, como su mismo nombre lo dice.

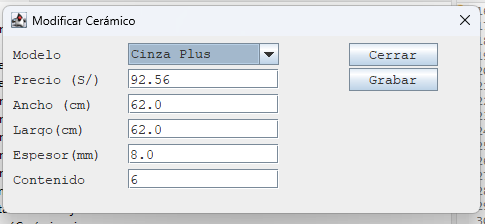




<https://i.imgur.com/ZVuO58X.jpg>

**Modificar Cerámicos**: La función de esta caja de diálogo consiste en que el cliente puede manipular la información de los productos y luego guardarlos temporalmente(hasta que cierre el JFrame principal).

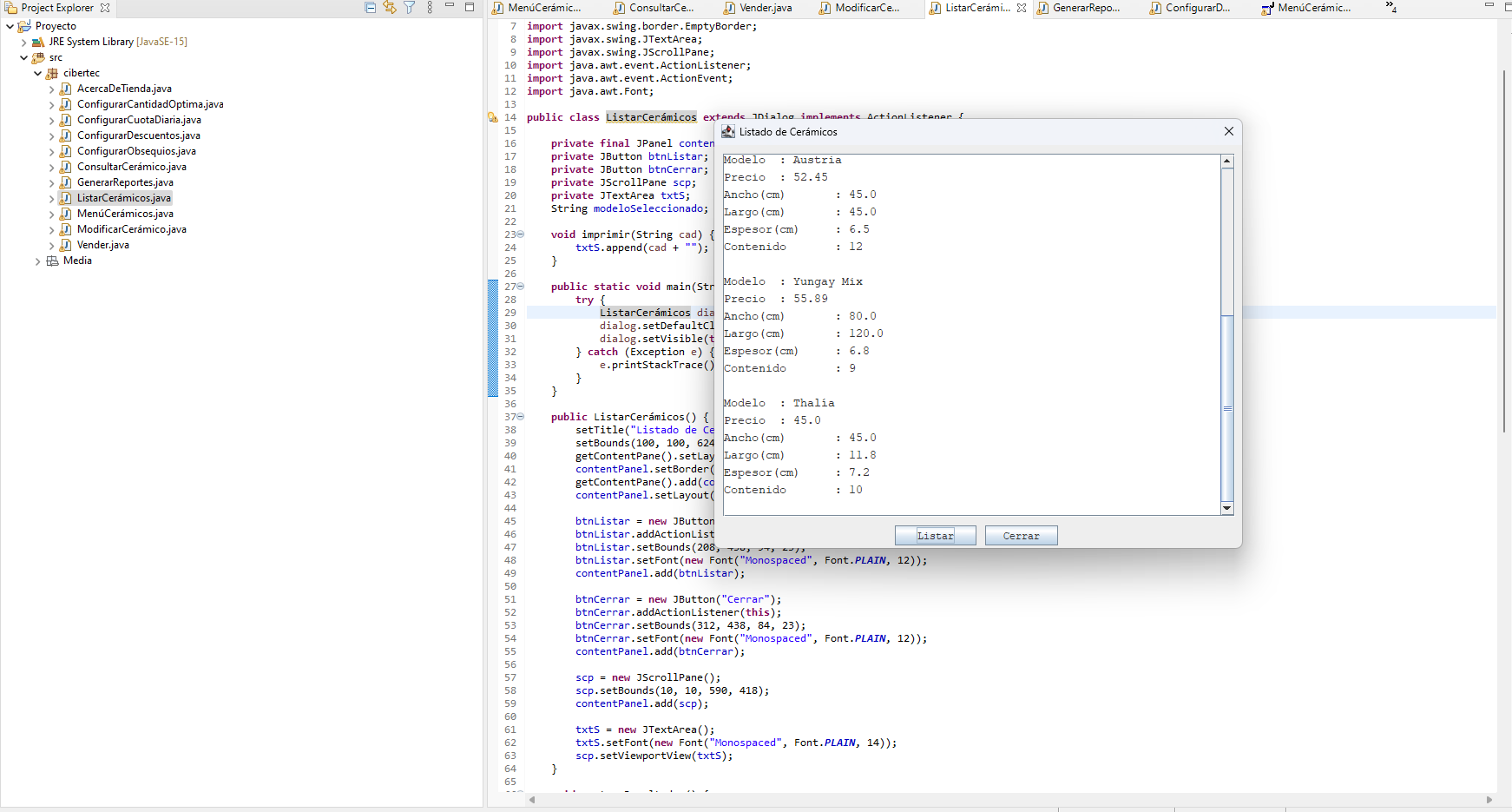


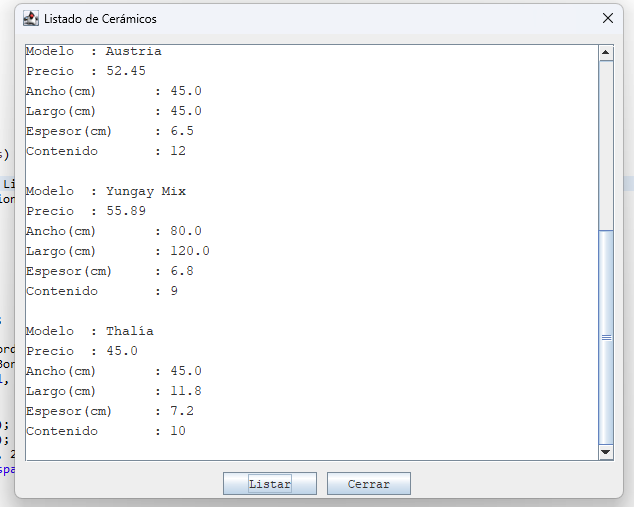


<https://i.imgur.com/bMotLis.jpg>

**Listado de Cerámicos:** La función de esta caja de diálogo consiste en que en al momento de que el usuario presione el botón “Listar” inmediatamente en el JTextArea

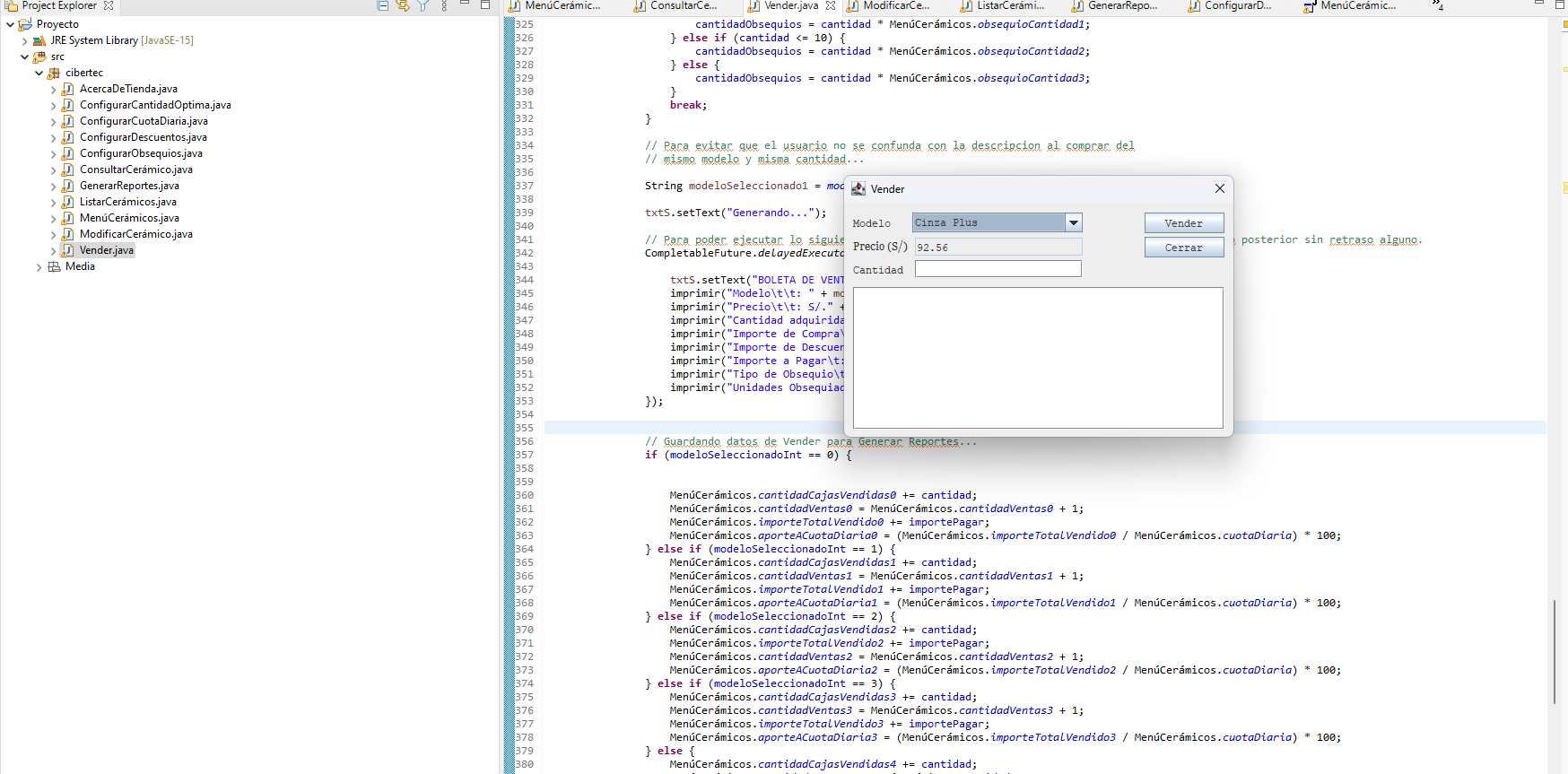
aparecerán los datos de cada Cerámico, ya sea que se muestren modificados o los que están por defecto.

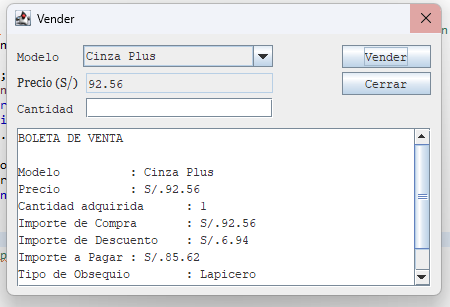




<https://i.imgur.com/hCXpxPu.jpg>

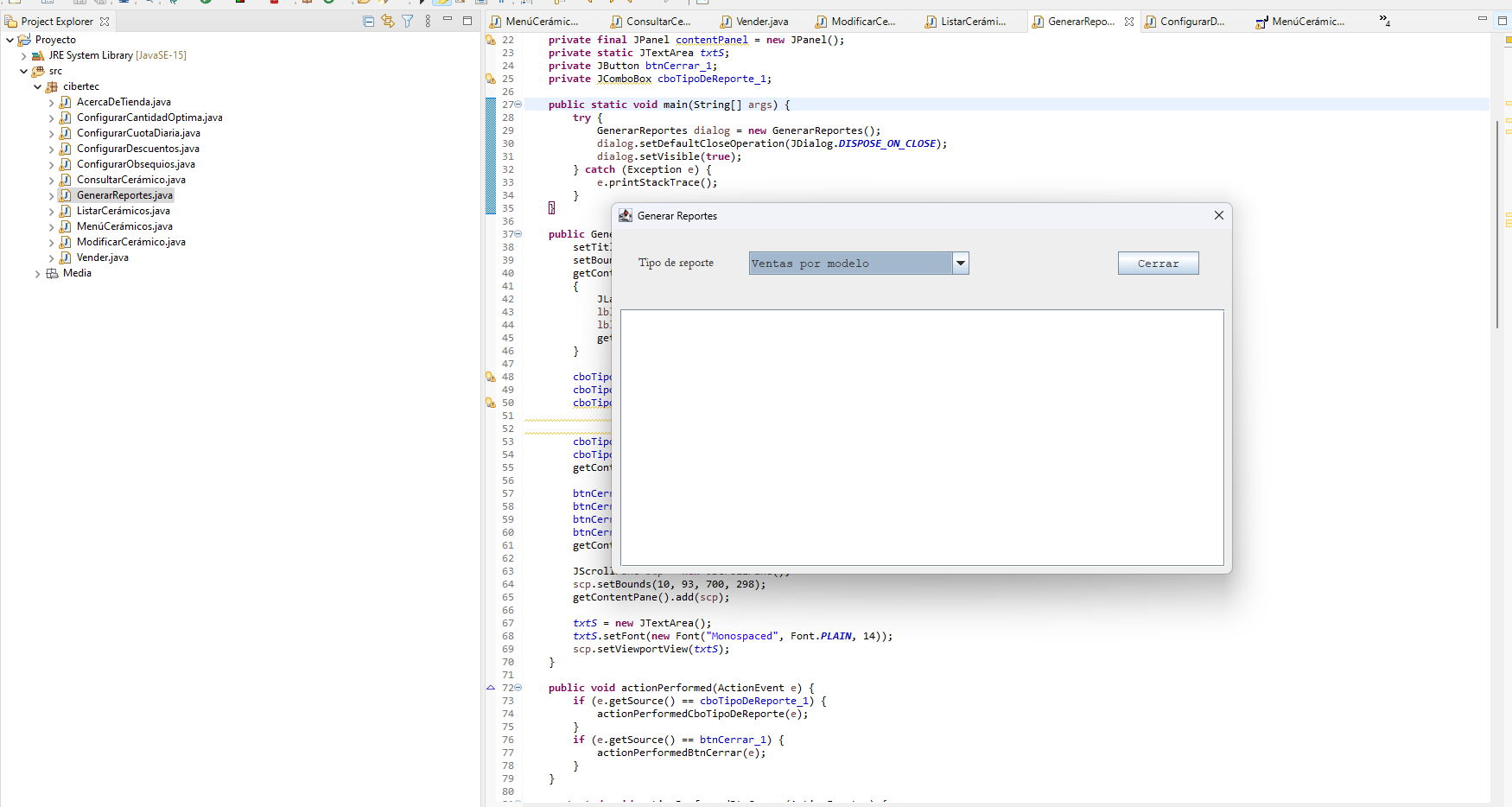
**Vender:** La función de esta caja de diálogo consiste en que el cliente pueda vender los productos y generar la boleta de venta. Mediante se supere cada 5 ventas, generará una pequeña ventana de información con el importe total general acumulado, más el porcentaje de la cuota diaria.

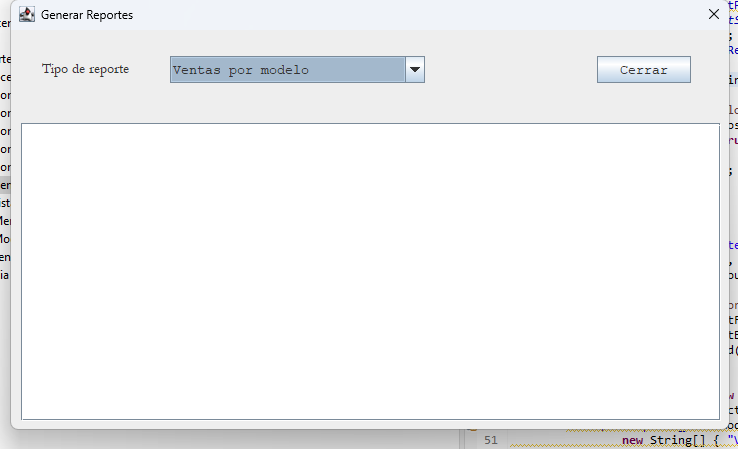




<https://i.imgur.com/wn57WFq.jpg>

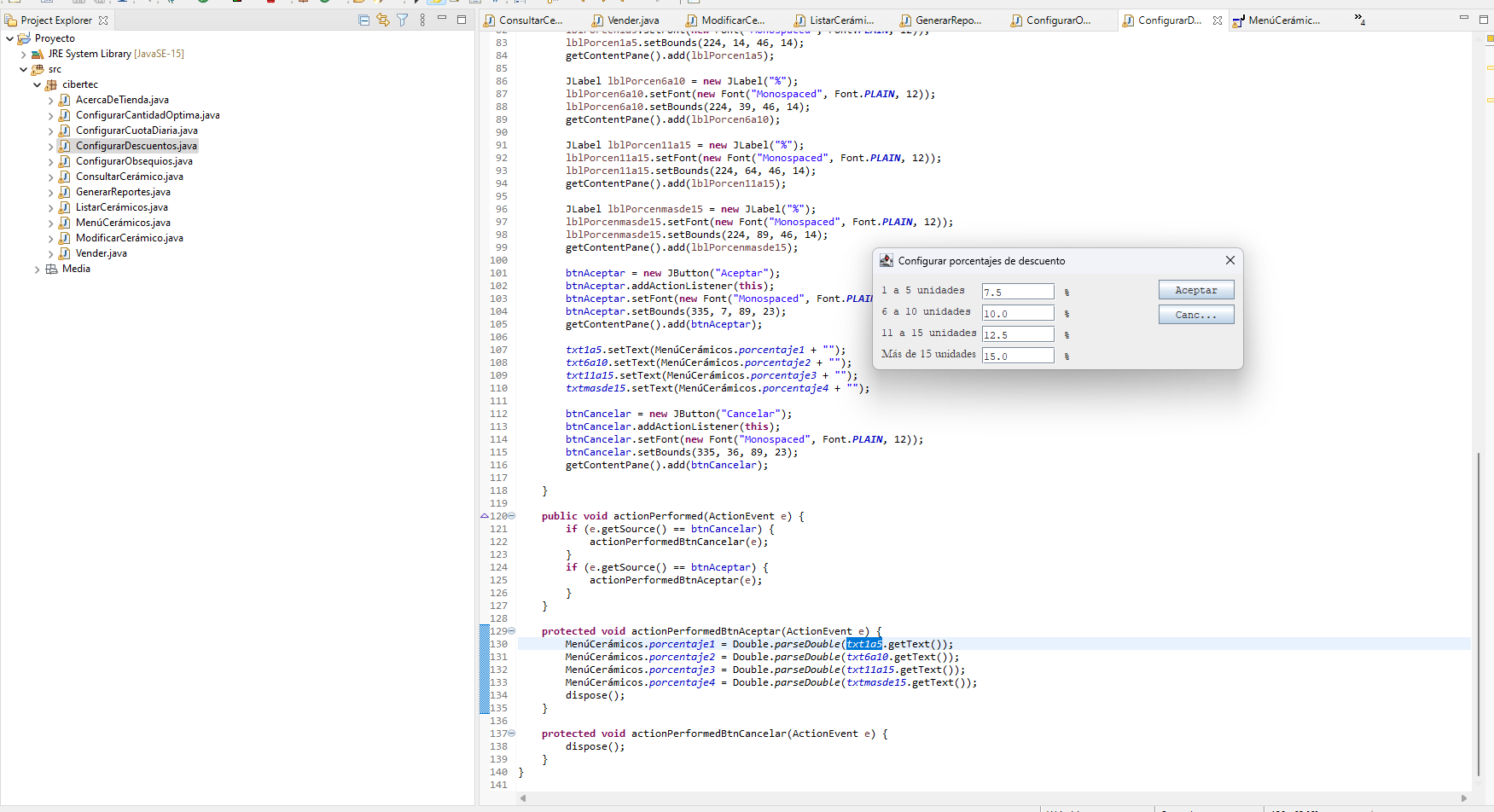
**Generar Reportes:** La función de esta caja de diálogo consiste en que el cliente pueda generar reportes de “Ventas por Modelo, Comparación de Precios con el precio promedio, Comparación de cajas vendidas con la cantidad óptima, Estadística sobre el precio”. Esta información podrá ser mostrada al instante cuando el cliente quiera generar algún reporte.

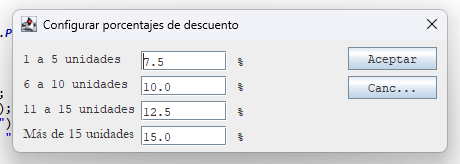




<https://i.imgur.com/PU33P5Y.jpg>

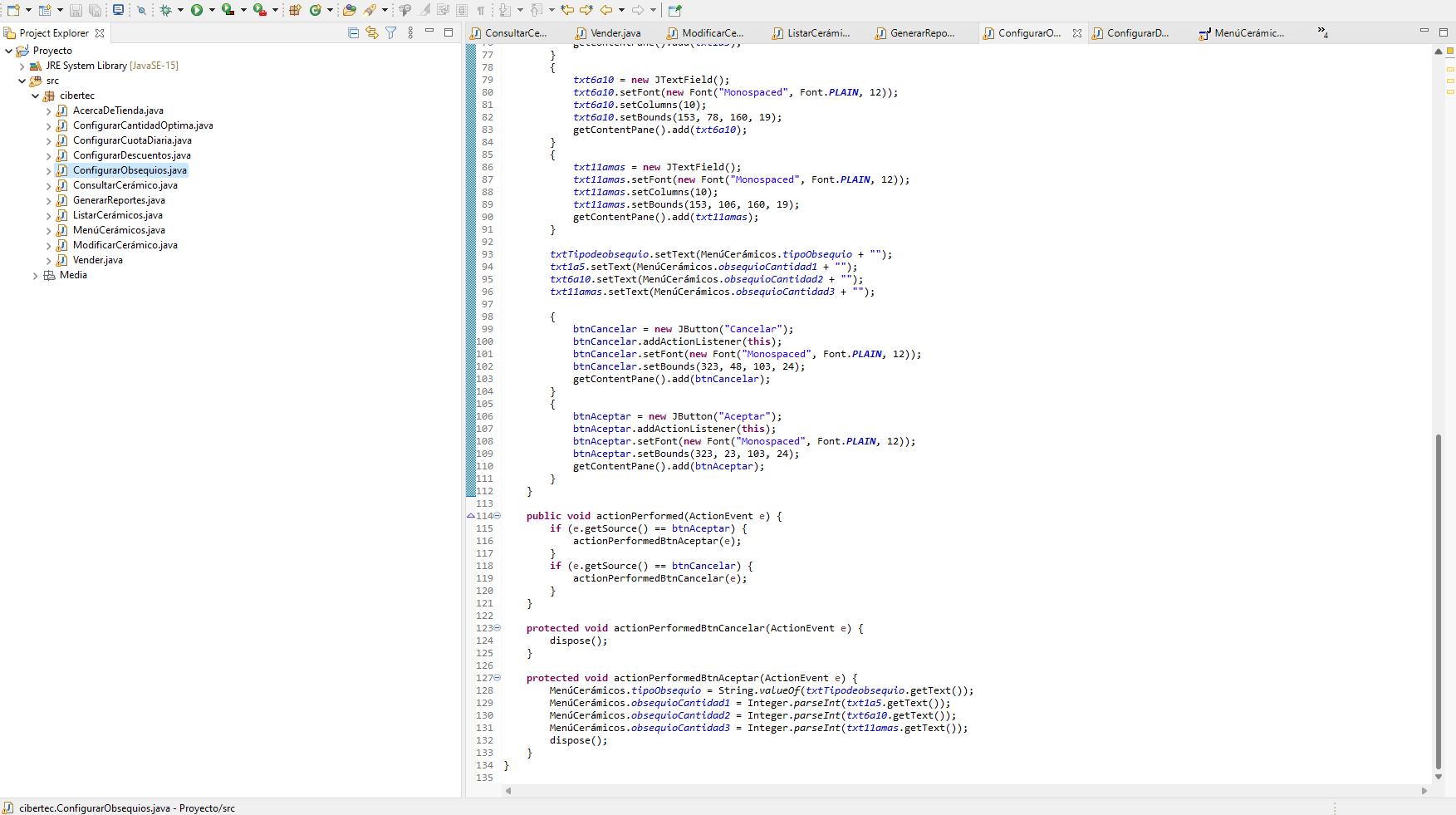
**Configurar Porcentaje de Descuentos:** La función de esta caja de diálogo consiste en que el cliente pueda modificar el porcentaje de descuento, dependiendo de la cantidad de unidades de sus productos. Esto se almacena en la variable global MenúCerámicos.*porcentaje1* a la vez se utiliza este mismo para mostrar el porcentaje actual.

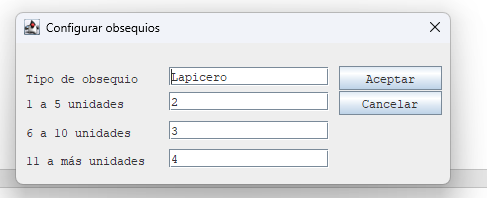




<https://i.imgur.com/HuETK3g.jpg>

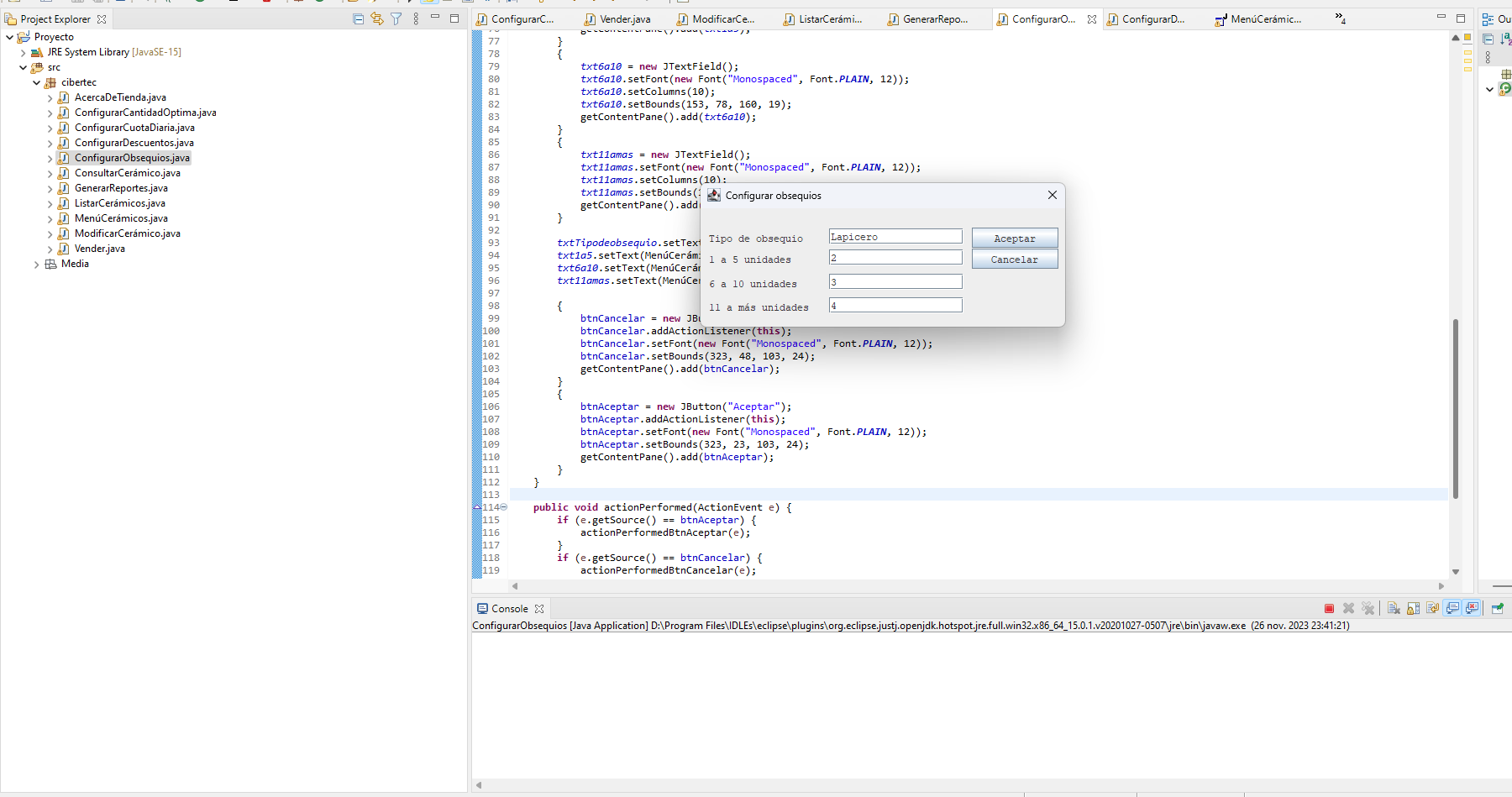
**Configurar Obsequios:** La función de esta caja de diálogo consiste en que el cliente pueda modificar el tipo de obsequio y la cantidad de unidades a obsequiar dependiendo de la cantidad de unidades. Esto se almacena en las variables globales, para poder guardar el resultado, con la misma se muestra.

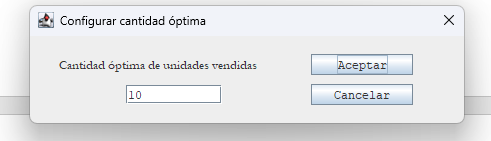




<https://i.imgur.com/qrE7MHo.jpg>

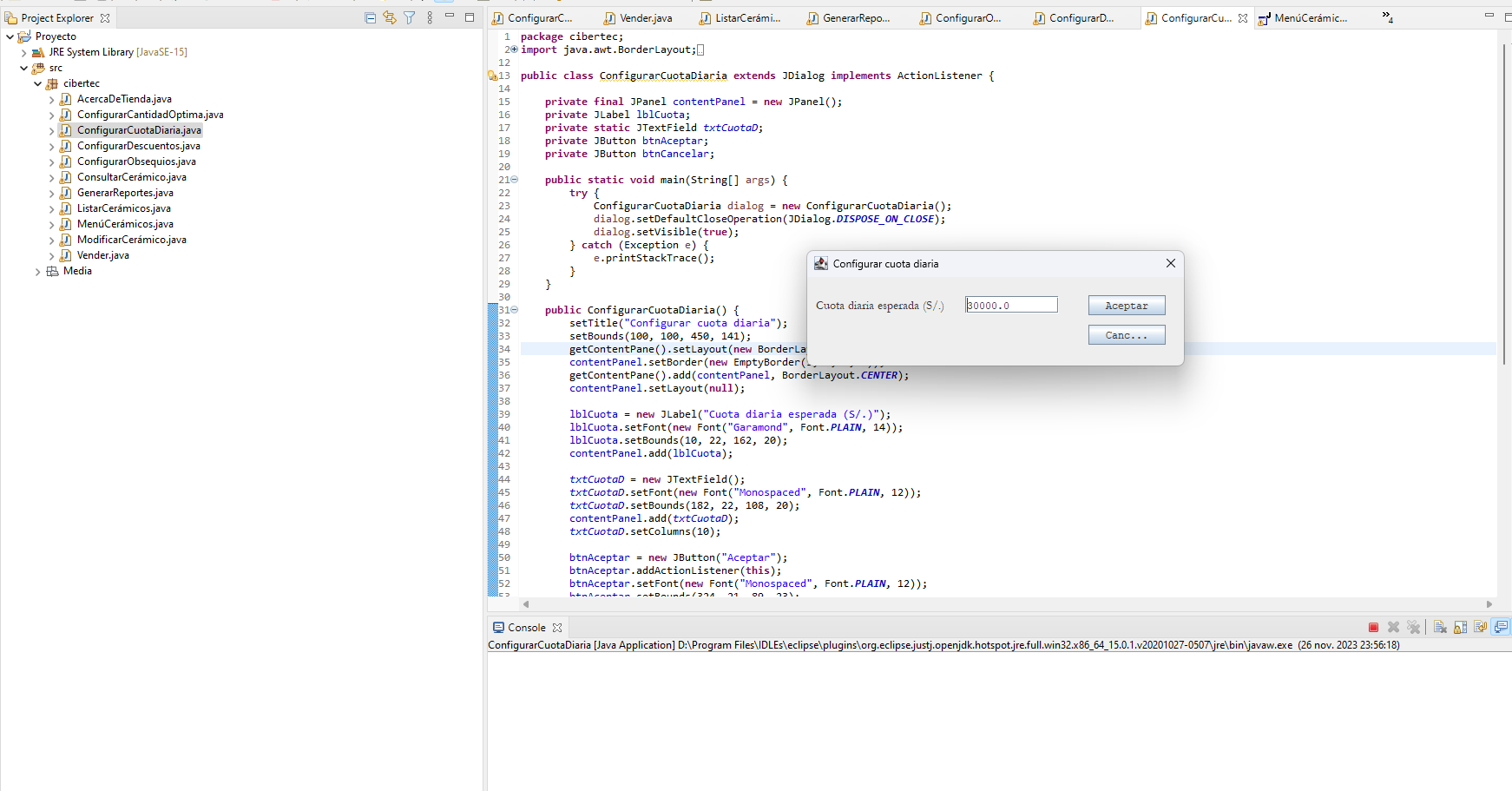
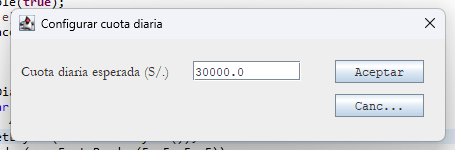
**Configurar cantidad Óptima:** La función de esta caja de diálogo consiste en que el cliente pueda configurar la cantidad óptima de unidades vendidas, se refleja el cambio en el apartado “Generar Reportes/COMPARACIÓN DE CAJAS VENDIDAS CON LA CANTIDAD ÓPTIMA”.





<https://i.imgur.com/oCrqu4a.jpg>

**Configurar cuota Diaria:** La función de esta caja de diálogo consiste en que el cliente pueda configurar la cantidad exacta que desea para la cuota Diara, siendo este una meta para sus empleados. Se almacena en la variable global y se utiliza la misma para mostrar.

<https://i.imgur.com/ESR7k6H.png>

#### Entregables

package cibertec;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import javax.swing.JTextField;

import java.awt.Font;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.DefaultComboBoxModel;

public class ModificarCerámico extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private static JTextField txtPrecio;

private static JTextField txtLargo;

private static JTextField txtEspesor;

private static JTextField txtContenido;

private static JTextField txtAncho;

private JButton btnCerrar;

private JButton btnGrabar;

private JComboBox cboModelo;

public static void main(String[] args) {

try {

ModificarCerámico dialog = new ModificarCerámico();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ModificarCerámico() {

setTitle("Modificar Cer\u00E1mico");

getContentPane().setFont(new Font("Dialog", Font.PLAIN, 15));

setBounds(100, 100, 459, 216);

getContentPane().setLayout(null);

JLabel lblModelo = new JLabel("Modelo");

lblModelo.setFont(new Font("arial", Font.PLAIN, 14));

lblModelo.setBounds(10, 13, 66, 14);

getContentPane().add(lblModelo);

JLabel lblPrecio = new JLabel("Precio (S/)");

lblPrecio.setFont(new Font("arial", Font.PLAIN, 14));

lblPrecio.setBounds(10, 40, 105, 14);

getContentPane().add(lblPrecio);

JLabel lblAncho = new JLabel("Ancho (cm)");

lblAncho.setFont(new Font("arial", Font.PLAIN, 14));

lblAncho.setBounds(10, 67, 94, 14);

getContentPane().add(lblAncho);

JLabel lblLargo = new JLabel("Largo(cm)");

lblLargo.setFont(new Font("arial", Font.PLAIN, 14));

lblLargo.setBounds(10, 94, 86, 14);

getContentPane().add(lblLargo);

JLabel lblEspesor = new JLabel("Espesor(mm)");

lblEspesor.setFont(new Font("arial", Font.PLAIN, 14));

lblEspesor.setBounds(10, 121, 94, 14);

getContentPane().add(lblEspesor);

JLabel lblContenido = new JLabel("Contenido");

lblContenido.setFont(new Font("arial", Font.PLAIN, 14));

lblContenido.setBounds(10, 148, 80, 14);

getContentPane().add(lblContenido);

cboModelo = new JComboBox();

cboModelo.addActionListener(this);

cboModelo.setModel(

new DefaultComboBoxModel(new String[] { "Cinza Plus", "Luxury", "Austria", "Yungay Mix", "Thalía" }));

cboModelo.setFont(new Font("arial", Font.PLAIN, 14));

cboModelo.setBounds(125, 7, 151, 22);

getContentPane().add(cboModelo);

txtPrecio = new JTextField();

txtPrecio.setFont(new Font("arial", Font.PLAIN, 14));

txtPrecio.setBounds(125, 36, 151, 20);

getContentPane().add(txtPrecio);

txtPrecio.setColumns(10);

txtAncho = new JTextField();

txtAncho.setFont(new Font("arial", Font.PLAIN, 14));

txtAncho.setBounds(125, 63, 151, 20);

getContentPane().add(txtAncho);

txtAncho.setColumns(10);

txtLargo = new JTextField();

txtLargo.setFont(new Font("arial", Font.PLAIN, 14));

txtLargo.setBounds(125, 90, 151, 20);

getContentPane().add(txtLargo);

txtLargo.setColumns(10);

txtEspesor = new JTextField();

txtEspesor.setFont(new Font("arial", Font.PLAIN, 14));

txtEspesor.setBounds(125, 117, 151, 20);

getContentPane().add(txtEspesor);

txtEspesor.setColumns(10);

txtContenido = new JTextField();

txtContenido.setFont(new Font("arial", Font.PLAIN, 14));

txtContenido.setBounds(125, 144, 151, 20);

getContentPane().add(txtContenido);

txtContenido.setColumns(10);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setFont(new Font("arial", Font.PLAIN, 14));

btnCerrar.setBounds(346, 9, 89, 23);

getContentPane().add(btnCerrar);

btnGrabar = new JButton("Grabar");

btnGrabar.addActionListener(this);

btnGrabar.setFont(new Font("arial", Font.PLAIN, 14));

btnGrabar.setBounds(346, 36, 89, 23);

getContentPane().add(btnGrabar);

txtPrecio.setText(MenúCerámicos.precio0 + "");

txtAncho.setText(MenúCerámicos.ancho0 + "");

txtLargo.setText(MenúCerámicos.largo0 + "");

txtEspesor.setText(MenúCerámicos.espesor0 + "");

txtContenido.setText(MenúCerámicos.contenido0 + "");

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == cboModelo) {

actionPerformedCboModelo(e);

}

if (e.getSource() == btnGrabar) {

actionPerformedBtnGrabar(e);

}

if (e.getSource() == btnCerrar) {

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

dispose();

}

//Lee los datos ingresados con getText y los guarda en las variable Locales de Menú Ceramico

protected void actionPerformedBtnGrabar(ActionEvent e) {

MenúCerámicos.precio0 = Double.parseDouble(txtPrecio.getText());

MenúCerámicos.ancho0 = Double.parseDouble(txtAncho.getText());

MenúCerámicos.largo0 = Double.parseDouble(txtLargo.getText());

MenúCerámicos.espesor0 = Double.parseDouble(txtEspesor.getText());

MenúCerámicos.contenido0 = (int) Double.parseDouble(txtContenido.getText());

dispose();

}

void mostrarDatos(double precio, double ancho, double largo, double espesor, int contenido) {

txtPrecio.setText(precio + "");

txtAncho.setText(ancho + "");

txtLargo.setText(largo + "");

txtEspesor.setText(espesor + "");

txtContenido.setText(contenido + "");

}

protected void actionPerformedCboModelo(ActionEvent e) {

switch (cboModelo.getSelectedIndex()) {

case 0:

mostrarDatos(MenúCerámicos.precio0, MenúCerámicos.ancho0, MenúCerámicos.largo0, MenúCerámicos.espesor0,MenúCerámicos.contenido0);

break;

case 1:

mostrarDatos(MenúCerámicos.precio1, MenúCerámicos.ancho1, MenúCerámicos.largo1, MenúCerámicos.espesor1,MenúCerámicos.contenido1);

break;

case 2:

mostrarDatos(MenúCerámicos.precio2, MenúCerámicos.ancho2, MenúCerámicos.largo2, MenúCerámicos.espesor2,MenúCerámicos.contenido2);

break;

case 3:

mostrarDatos(MenúCerámicos.precio3, MenúCerámicos.ancho3, MenúCerámicos.largo3, MenúCerámicos.espesor3,MenúCerámicos.contenido3);

break;

default:

mostrarDatos(MenúCerámicos.precio4, MenúCerámicos.ancho4, MenúCerámicos.largo4, MenúCerámicos.espesor4,MenúCerámicos.contenido4);

break;

}

}

}

package cibertec;

import java.awt.FlowLayout;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JComboBox;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JTextField;

import javax.swing.JTextArea;

import java.awt.event.ActionListener;

import java.text.DecimalFormat;

import java.util.concurrent.CompletableFuture;

import java.util.concurrent.TimeUnit;

import java.awt.event.ActionEvent;

import javax.swing.JScrollPane;

import java.awt.Font;

public class Vender extends JDialog {

static String modeloSeleccionado;

int modeloSeleccionadoInt;

int acumuladorVender = 0;

double importeTotalGeneralAcumulado = 0;

DecimalFormat df = new DecimalFormat("0.00"); // para usar el formatear números decimales, se debe importar (import

// java.text.DecimalFormat)

double importeCompra = 0, importeDescuento = 0, precioSeleccionado = 0, importePagar = 0, porcentajeCuotaDiaria = 0;

int cantidad, cantidadObsequios = 0;

private JTextField txtCantidad;

private static JTextField txtPrecio;

private final JPanel contentPanel = new JPanel();

public static void main(String[] args) {

try {

Vender dialog = new Vender();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public Vender() {

setTitle("Vender");

setBounds(100, 100, 450, 300);

getContentPane().setLayout(null);

contentPanel.setBounds(0, 0, 0, 0);

contentPanel.setLayout(new FlowLayout());

contentPanel.setBorder(new EmptyBorder(5, 5, 5, 5));

getContentPane().add(contentPanel);

JButton btnVender = new JButton("Vender");

btnVender.setFont(new Font("arial", Font.PLAIN, 12));

btnVender.setBounds(335, 11, 89, 23);

getContentPane().add(btnVender);

JButton btnCerrar = new JButton("Cerrar");

btnCerrar.setFont(new Font("arial", Font.PLAIN, 12));

btnCerrar.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

dispose();

}

});

btnCerrar.setBounds(335, 38, 89, 23);

getContentPane().add(btnCerrar);

JLabel lblModelo = new JLabel("Modelo");

lblModelo.setFont(new Font("arial", Font.PLAIN, 12));

lblModelo.setBounds(10, 15, 46, 14);

getContentPane().add(lblModelo);

JLabel lblPrecio = new JLabel("Precio (S/)");

lblPrecio.setFont(new Font("Georgia", Font.PLAIN, 12));

lblPrecio.setBounds(10, 42, 68, 14);

getContentPane().add(lblPrecio);

JLabel lblCantidad = new JLabel("Cantidad");

lblCantidad.setFont(new Font("arial", Font.PLAIN, 12));

lblCantidad.setBounds(10, 67, 59, 14);

getContentPane().add(lblCantidad);

JComboBox cboModelo = new JComboBox();

cboModelo.setFont(new Font("arial", Font.PLAIN, 12));

cboModelo.setToolTipText("Selecciona un tipo");

cboModelo.setModel(

new DefaultComboBoxModel(new String[] { "Cinza Plus", "Luxury", "Austria", "Yungay Mix", "Thalía" }));

cboModelo.setSelectedIndex(0);

cboModelo.setBounds(76, 11, 190, 22);

getContentPane().add(cboModelo);

txtCantidad = new JTextField();

txtCantidad.setFont(new Font("arial", Font.PLAIN, 12));

txtCantidad.setBounds(79, 64, 187, 20);

getContentPane().add(txtCantidad);

txtCantidad.setColumns(10);

txtPrecio = new JTextField();

txtPrecio.setFont(new Font("arial", Font.PLAIN, 12));

txtPrecio.setEditable(false);

txtPrecio.setColumns(10);

txtPrecio.setBounds(79, 39, 187, 20);

getContentPane().add(txtPrecio);

JScrollPane scp = new JScrollPane();

scp.setBounds(10, 94, 414, 159);

getContentPane().add(scp);

JTextArea txtS = new JTextArea();

txtS.setFont(new Font("arial", Font.PLAIN, 12));

scp.setViewportView(txtS);

// Valores de Modelo0 por default...

txtPrecio.setText(MenúCerámicos.precio0 + "");

modeloSeleccionado = "0";

cboModelo.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

// Código a ejecutar cuando se selecciona un elemento en cboModelo

switch (cboModelo.getSelectedIndex()) {

case 0:

txtPrecio.setText(MenúCerámicos.precio0 + "");

modeloSeleccionadoInt = 0;

modeloSeleccionado = "0";

break;

case 1:

txtPrecio.setText(MenúCerámicos.precio1 + "");

modeloSeleccionadoInt = 1;

modeloSeleccionado = "1";

break;

case 2:

txtPrecio.setText(MenúCerámicos.precio2 + "");

modeloSeleccionadoInt = 2;

modeloSeleccionado = "2";

break;

case 3:

txtPrecio.setText(MenúCerámicos.precio3 + "");

modeloSeleccionadoInt = 3;

modeloSeleccionado = "3";

break;

case 4:

txtPrecio.setText(MenúCerámicos.precio4 + "");

modeloSeleccionadoInt = 4;

modeloSeleccionado = "4";

break;

}

}

});

btnVender.addActionListener(new ActionListener() {

// -----------------------------------------------------------------------------------------

// Metodos

void calcularImporteDescuento() {

if (cantidad <= 5) {

importeDescuento = (MenúCerámicos.porcentaje1 \* importeCompra) / 100;

} else if (cantidad <= 10) {

importeDescuento = (MenúCerámicos.porcentaje2 \* importeCompra) / 100;

} else if (cantidad <= 15) {

importeDescuento = (MenúCerámicos.porcentaje3 \* importeCompra) / 100;

} else {

importeDescuento = (MenúCerámicos.porcentaje4 \* importeCompra) / 100;

}

}

void calcularImporteCompra(double precio) {

importeCompra = precio \* cantidad;

}

void calcularImportePagar() {

importePagar = importeCompra - importeDescuento;

}

void calcularObsequios() {

if (cantidad <= 5) {

cantidadObsequios = cantidad \* MenúCerámicos.obsequioCantidad1;

} else if (cantidad <= 10) {

cantidadObsequios = cantidad \* MenúCerámicos.obsequioCantidad2;

} else {

cantidadObsequios = cantidad \* MenúCerámicos.obsequioCantidad3;

}

}

void guardarDatosGenerar(int cantidadCajas, int cantidadVentas, double importeTotalVendido, double aporteCuota) {

cantidadCajas += cantidad;

cantidadVentas += 1;

importeTotalVendido += importePagar;

aporteCuota= (importeTotalVendido / MenúCerámicos.cuotaDiaria)

\* 100;

}

// ----------------------------------------------------

public void actionPerformed(ActionEvent e) {

// Código a ejecutar cuando se selecciona un elemento en btnVender

// Al poner Vender verifica si la cantidad introducida es un numero

if (!isNumeric(txtCantidad.getText())) {

txtS.setText("Debe introducir un número");

return; // para salir del método devolviendo un valor o no (en este caso no.)

}

cantidad = Integer.parseInt(txtCantidad.getText());

if (cantidad < 1) {

txtS.setText("Debe introducir un número mayor a 0");

return;

}

// Saber que modelo es

switch (modeloSeleccionado) {

case "0":

modeloSeleccionado = MenúCerámicos.modelo0;

precioSeleccionado = MenúCerámicos.precio0;

calcularImporteCompra(MenúCerámicos.precio0);

calcularImporteDescuento();

calcularImportePagar();

calcularObsequios();

break;

case "1":

modeloSeleccionado = MenúCerámicos.modelo1;

precioSeleccionado = MenúCerámicos.precio1;

calcularImporteCompra(MenúCerámicos.precio1);

calcularImporteDescuento();

calcularImportePagar();

calcularObsequios();

break;

case "2":

modeloSeleccionado = MenúCerámicos.modelo2;

precioSeleccionado = MenúCerámicos.precio2;

calcularImporteCompra(MenúCerámicos.precio2);

calcularImporteDescuento();

calcularImportePagar();

calcularObsequios();

break;

case "3":

modeloSeleccionado = MenúCerámicos.modelo3;

precioSeleccionado = MenúCerámicos.precio3;

calcularImporteCompra(MenúCerámicos.precio3);

calcularImporteDescuento();

calcularImportePagar();

calcularObsequios();

break;

case "4":

modeloSeleccionado = MenúCerámicos.modelo4;

precioSeleccionado = MenúCerámicos.precio4;

calcularImporteCompra(MenúCerámicos.precio4);

calcularImporteDescuento();

calcularImportePagar();

calcularObsequios();

break;

}

// Para evitar que el usuario no se confunda con la descripcion al comprar del

// mismo modelo y misma cantidad...

String modeloSeleccionado1 = modeloSeleccionado;

txtS.setText("Generando...");

// Para poder ejecutar lo siguiente luego de 0.5 segundos. mientras sigue

// ejecutando el código posterior sin retraso alguno.

CompletableFuture.delayedExecutor(500, TimeUnit.MILLISECONDS).execute(() -> {

txtS.setText("BOLETA DE VENTA" + "\n\n");

txtS.append("Modelo\t\t: " + modeloSeleccionado1 + "\n");

txtS.append("Precio\t\t: S/." + df.format(precioSeleccionado) + "\n");

txtS.append("Cantidad adquirida\t: " + cantidad + "\n");

txtS.append("Importe de Compra\t: S/." + df.format(importeCompra) + "\n");

txtS.append("Importe de Descuento\t: S/." + df.format(importeDescuento) + "\n");

txtS.append("Importe a Pagar\t: S/." + df.format(importePagar) + "\n");

txtS.append("Tipo de Obsequio\t: " + MenúCerámicos.tipoObsequio + "\n");

txtS.append("Unidades Obsequiadas\t: " + cantidadObsequios + "\n");

});

// Guardando datos de Vender en las variables globales para Generar Reportes...

if (modeloSeleccionadoInt == 0) {

guardarDatosGenerar(MenúCerámicos.cantidadCajasVendidas0,MenúCerámicos.cantidadVentas0,MenúCerámicos.importeTotalVendido0,MenúCerámicos.aporteACuotaDiaria0);

} else if (modeloSeleccionadoInt == 1) {

MenúCerámicos.cantidadCajasVendidas1 += cantidad;

MenúCerámicos.cantidadVentas1 += 1;

MenúCerámicos.importeTotalVendido1 += importePagar;

MenúCerámicos.aporteACuotaDiaria1 = (MenúCerámicos.importeTotalVendido1 / MenúCerámicos.cuotaDiaria)

\* 100;

} else if (modeloSeleccionadoInt == 2) {

MenúCerámicos.cantidadCajasVendidas2 += cantidad;

MenúCerámicos.importeTotalVendido2 += importePagar;

MenúCerámicos.cantidadVentas2 += 1;

MenúCerámicos.aporteACuotaDiaria2 = (MenúCerámicos.importeTotalVendido2 / MenúCerámicos.cuotaDiaria)

\* 100;

} else if (modeloSeleccionadoInt == 3) {

MenúCerámicos.cantidadCajasVendidas3 += cantidad;

MenúCerámicos.cantidadVentas3 += 1;

MenúCerámicos.importeTotalVendido3 += importePagar;

MenúCerámicos.aporteACuotaDiaria3 = (MenúCerámicos.importeTotalVendido3 / MenúCerámicos.cuotaDiaria)

\* 100;

} else {

MenúCerámicos.cantidadCajasVendidas4 += cantidad;

MenúCerámicos.cantidadVentas4 += 1;

MenúCerámicos.importeTotalVendido4 += importePagar;

MenúCerámicos.aporteACuotaDiaria4 = (MenúCerámicos.importeTotalVendido4 / MenúCerámicos.cuotaDiaria)

\* 100;

}

acumuladorVender++;

// Calcular Importe General Acumulado y Porcentaje de la cuota diaria

importeTotalGeneralAcumulado = importeTotalGeneralAcumulado + importePagar;

porcentajeCuotaDiaria = (importeTotalGeneralAcumulado / MenúCerámicos.cuotaDiaria) \* 100;

// Alerta cada 5 ventas

if (acumuladorVender % 5 == 0) {

JOptionPane.showConfirmDialog(null,

"Venta Nro. " + acumuladorVender + "\nImporte total general acumulado: S/. "

+ df.format(importeTotalGeneralAcumulado) + "\nPorcentaje de la cuota diaria: "

+ df.format(porcentajeCuotaDiaria) + "%",

"Avance de ventas", JOptionPane.OK\_OPTION);

}

// Reiniciar la selección - Todo

cboModelo.setSelectedIndex(0);

txtCantidad.setText("");

}

private boolean isNumeric(String text) { // Private (para poder usar solo en esta clase) boolean (verdadero

// o falso) isNumeric (función)

try {

Integer.parseInt(text); // Convertir la cadena a un número entero

return true; // Si la conversión tiene éxito, retorna true

} catch (Exception e) {

return false; // Si ocurre alguna excepción durante la conversión, retorna false

}

}

});

}

}

package cibertec;

import java.awt.EventQueue;

import javax.swing.JOptionPane;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JMenuBar;

import javax.swing.JMenu;

import javax.swing.JMenuItem;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.Font;

import javax.swing.ImageIcon;

public class MenúCerámicos extends JFrame implements ActionListener {

private JPanel contentPane;

// Datos mínimos del primer producto

public static String modelo0 = "Cinza Plus";

public static double precio0 = 92.56;

public static double ancho0 = 62.0;

public static double largo0 = 62.0;

public static double espesor0 = 8;

public static int contenido0 = 6;

// Datos mínimos del segundo producto

public static String modelo1 = "Luxury";

public static double precio1 = 42.77;

public static double ancho1 = 60;

public static double largo1 = 60;

public static double espesor1 = 8.5;

public static int contenido1 = 4;

// Datos mínimos del tercer producto

public static String modelo2 = "Austria";

public static double precio2 = 52.45;

public static double ancho2 = 45;

public static double largo2 = 45;

public static double espesor2 = 6.5;

public static int contenido2 = 12;

// Datos mínimos del cuarto producto

public static String modelo3 = "Yungay Mix";

public static double precio3 = 55.89;

public static double ancho3 = 80;

public static double largo3 = 120;

public static double espesor3 = 6.8;

public static int contenido3 = 9;

// Datos mínimos del quinto producto

public static String modelo4 = "Thalía";

public static double precio4 = 45;

public static double ancho4 = 45;

public static double largo4 = 11.8;

public static double espesor4 = 7.2;

public static int contenido4 = 10;

// Porcentajes de descuento

public static double porcentaje1 = 7.5;

public static double porcentaje2 = 10.0;

public static double porcentaje3 = 12.5;

public static double porcentaje4 = 15.0;

// Obsequio

public static String tipoObsequio = "Lapicero";

public static int obsequioCantidad1 = 2;

public static int obsequioCantidad2 = 3;

public static int obsequioCantidad3 = 4;

// Cantidad óptima de unidades vendidas por día

public static int cantidadOptima = 10;

// Cuota diaria

public static double cuotaDiaria = 30000;

//Variables Totales de GenerarReportes

public static int cantidadVentas0;

public static int cantidadVentas1;

public static int cantidadVentas2;

public static int cantidadVentas3;

public static int cantidadVentas4;

public static int cantidadCajasVendidas0;

public static int cantidadCajasVendidas1;

public static int cantidadCajasVendidas2;

public static int cantidadCajasVendidas3;

public static int cantidadCajasVendidas4;

public static double importeTotalVendido0;

public static double importeTotalVendido1;

public static double importeTotalVendido2;

public static double importeTotalVendido3;

public static double importeTotalVendido4;

public static double aporteACuotaDiaria0;

public static double aporteACuotaDiaria1;

public static double aporteACuotaDiaria2;

public static double aporteACuotaDiaria3;

public static double aporteACuotaDiaria4;

// para poder obtener el promedio de los precios para "Ventas/GenerarReportes" en la pestaña de estadisticas.

public static double precioProm=(precio0+precio1+precio2+precio3+precio4)/5;

private JMenuItem mntmSalir;

private JMenuItem mntmConsultarceramico;

private JMenuItem mntmModificarceramico;

private JMenuItem mntmListarceramico;

private JMenuItem mntmVender;

private JMenuItem mntmGenerarreportes;

private JMenuItem mntmConfigurardescuentos;

private JMenuItem mntmConfigurarobsequios;

private JMenuItem mntmConfigurarcantidadoptima;

private JMenuItem mntmConfigurarcuotadiaria;

private JMenuItem mntmAcerdadetienda;

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

MenúCerámicos frame = new MenúCerámicos();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

public MenúCerámicos() {

setTitle("Venta de Cerámicos");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 646, 434);

JMenuBar menuBar = new JMenuBar();

menuBar.setFont(new Font("arial", Font.PLAIN, 12));

setJMenuBar(menuBar);

JMenu mnArchivo = new JMenu("Archivo");

mnArchivo.setFont(new Font("arial", Font.PLAIN, 12));

menuBar.add(mnArchivo);

mntmSalir = new JMenuItem("Salir");

mntmSalir.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/boton-x.png")));

mntmSalir.addActionListener(this);

mntmSalir.setFont(new Font("arial", Font.PLAIN, 12));

mnArchivo.add(mntmSalir);

JMenu mnMantenimiento = new JMenu("Mantenimiento");

mnMantenimiento.setFont(new Font("arial", Font.PLAIN, 12));

menuBar.add(mnMantenimiento);

mntmConsultarceramico = new JMenuItem("Consultar cerámico");

mntmConsultarceramico.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/consulta.png")));

mntmConsultarceramico.addActionListener(this);

mntmConsultarceramico.setFont(new Font("arial", Font.PLAIN, 12));

mnMantenimiento.add(mntmConsultarceramico);

mntmModificarceramico = new JMenuItem("Modificar cerámico");

mntmModificarceramico.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/editar.png")));

mntmModificarceramico.addActionListener(this);

mntmModificarceramico.setFont(new Font("arial", Font.PLAIN, 12));

mnMantenimiento.add(mntmModificarceramico);

mntmListarceramico = new JMenuItem("Listar cerámicos");

mntmListarceramico.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/lista-de-verificacion.png")));

mntmListarceramico.addActionListener(this);

mntmListarceramico.setFont(new Font("arial", Font.PLAIN, 12));

mnMantenimiento.add(mntmListarceramico);

JMenu mnVentas = new JMenu("Ventas");

mnVentas.setFont(new Font("arial", Font.PLAIN, 12));

menuBar.add(mnVentas);

mntmVender = new JMenuItem("Vender");

mntmVender.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/tienda.png")));

mntmVender.addActionListener(this);

mntmVender.setFont(new Font("arial", Font.PLAIN, 12));

mnVentas.add(mntmVender);

mntmGenerarreportes = new JMenuItem("Generar reportes");

mntmGenerarreportes.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/reporte-de-negocios.png")));

mntmGenerarreportes.addActionListener(this);

mntmGenerarreportes.setFont(new Font("arial", Font.PLAIN, 12));

mnVentas.add(mntmGenerarreportes);

JMenu mnConfiguracion = new JMenu("Configuración");

mnConfiguracion.setFont(new Font("arial", Font.PLAIN, 12));

menuBar.add(mnConfiguracion);

mntmConfigurardescuentos = new JMenuItem("Configurar descuentos");

mntmConfigurardescuentos.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/descuento.png")));

mntmConfigurardescuentos.addActionListener(this);

mntmConfigurardescuentos.setFont(new Font("arial", Font.PLAIN, 12));

mnConfiguracion.add(mntmConfigurardescuentos);

mntmConfigurarobsequios = new JMenuItem("Configurar obsequios");

mntmConfigurarobsequios.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/obsequio.png")));

mntmConfigurarobsequios.addActionListener(this);

mntmConfigurarobsequios.setFont(new Font("arial", Font.PLAIN, 12));

mnConfiguracion.add(mntmConfigurarobsequios);

mntmConfigurarcantidadoptima = new JMenuItem("Configurar cantidad óptima");

mntmConfigurarcantidadoptima.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/desbordamiento.png")));

mntmConfigurarcantidadoptima.addActionListener(this);

mntmConfigurarcantidadoptima.setFont(new Font("arial", Font.PLAIN, 12));

mnConfiguracion.add(mntmConfigurarcantidadoptima);

mntmConfigurarcuotadiaria = new JMenuItem("Configurar cuota diaria");

mntmConfigurarcuotadiaria.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/rutina.png")));

mntmConfigurarcuotadiaria.addActionListener(this);

mntmConfigurarcuotadiaria.setFont(new Font("arial", Font.PLAIN, 12));

mnConfiguracion.add(mntmConfigurarcuotadiaria);

JMenu mnAyuda = new JMenu("Ayuda");

mnAyuda.setFont(new Font("arial", Font.PLAIN, 12));

menuBar.add(mnAyuda);

mntmAcerdadetienda = new JMenuItem("Acerca de Tienda");

mntmAcerdadetienda.setIcon(new ImageIcon(MenúCerámicos.class.getResource("/Media/pregunta.png")));

mntmAcerdadetienda.addActionListener(this);

mntmAcerdadetienda.setFont(new Font("arial", Font.PLAIN, 12));

mnAyuda.add(mntmAcerdadetienda);

contentPane = new JPanel();

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

setContentPane(contentPane);

contentPane.setLayout(null);

}

public void actionPerformed(ActionEvent e) { // eventos para uso de interacción

if (e.getSource() == mntmAcerdadetienda) { // comprobar sí e.getSource() es igual al objeto mntmAcercadadetienda

actionPerformedMntmAcerdadetienda(e); // se le pasa el evento "e" a actionPerfomedMntmAcercadadetienda para poder manejar las acciones luego de la interacción.

}

if (e.getSource() == mntmConfigurarcuotadiaria) {

actionPerformedMntmConfigurarcuotadiaria(e);

}

if (e.getSource() == mntmConfigurarcantidadoptima) {

actionPerformedMntmConfigurarcantidadoptima(e);

}

if (e.getSource() == mntmConfigurarobsequios) {

actionPerformedMntmConfigurarobsequios(e);

}

if (e.getSource() == mntmConfigurardescuentos) {

actionPerformedMntmConfigurardescuentos(e);

}

if (e.getSource() == mntmGenerarreportes) {

actionPerformedMntmGenerarreportes(e);

}

if (e.getSource() == mntmVender) {

actionPerformedMntmVender(e);

}

if (e.getSource() == mntmListarceramico) {

actionPerformedMntmListarceramico(e);

}

if (e.getSource() == mntmModificarceramico) {

actionPerformedMntmModificarceramico(e);

}

if (e.getSource() == mntmConsultarceramico) {

actionPerformedMntmConsultarceramico(e);

}

if (e.getSource() == mntmSalir) {

actionPerformedMntmSalir(e);

}

}

protected void actionPerformedMntmSalir(ActionEvent e) { // Acción del Salir

// Importamos esto "import javax.swing.JOptionPane;" para poder utilizar la función de JOptionPane.showConfirmDialog

/\*

OptionPane.showConfirmDialog(Component parentComponent, -- Componente sobre el que se abre el cuadro de diálogo (this para el componente actual)

Object message, -- El mensaje

String title, -- Titulo

int optionType, -- OK\_CANCEL\_OPTION(Botones Aceptar y Cancelar), CLOSED\_OPTION(Botón Aceptar), YES\_NO\_OPTION(Botones Sí y No), YES\_NO\_CANCEL\_OPTION(Botones Sí, No y Cancelar)

int messageType, -- optional INFORMATION\_MESSAGE. Icono de información. WARNING\_MESSAGE. Icono de advertencia.ERROR\_MESSAGE. Icono de error QUESTION\_MESSAGE. Icono de pregunta. PLAIN\_MESSAGE. Sin icono DEFAULT\_OPTION. Sin icono.

Icon icon); -- optional Url del icono (necesario estar dentro del package)

\*

\* \*/

if (JOptionPane.showConfirmDialog(this, "¿Está seguro que desea salir?", "Salir",JOptionPane.YES\_OPTION) == JOptionPane.YES\_OPTION) {

System.exit(0); // Para poder cerrar la aplicación completamente

}

}

protected void actionPerformedMntmConsultarceramico(ActionEvent e) { // la acción de apertura de ventana

ConsultarCerámico d = new ConsultarCerámico(); // se inicia la instancia y se le asigna a la variable "d"

d.setLocationRelativeTo(mntmConsultarceramico); // para ubicar la ventana de otro componente "mntmConsultarceramico"

d.setVisible(true); // poder hacer visible la ventana ubicada de "mntmConsultarceramico"

}

protected void actionPerformedMntmModificarceramico(ActionEvent e) {

ModificarCerámico d = new ModificarCerámico();

d.setLocationRelativeTo(mntmModificarceramico);

d.setVisible(true);

}

protected void actionPerformedMntmListarceramico(ActionEvent e) {

ListarCerámicos d = new ListarCerámicos();

d.setLocationRelativeTo(mntmListarceramico);

d.setVisible(true);

}

protected void actionPerformedMntmVender(ActionEvent e) {

Vender d = new Vender();

d.setLocationRelativeTo(mntmListarceramico);

d.setVisible(true);

}

protected void actionPerformedMntmGenerarreportes(ActionEvent e) {

GenerarReportes d = new GenerarReportes();

d.setLocationRelativeTo(mntmListarceramico);

d.setVisible(true);

}

protected void actionPerformedMntmConfigurardescuentos(ActionEvent e) {

ConfigurarDescuentos d = new ConfigurarDescuentos();

d.setLocationRelativeTo(mntmConfigurardescuentos);

d.setVisible(true);

}

protected void actionPerformedMntmConfigurarobsequios(ActionEvent e) {

ConfigurarObsequios d = new ConfigurarObsequios();

d.setLocationRelativeTo(mntmConfigurarobsequios);

d.setVisible(true);

}

protected void actionPerformedMntmConfigurarcantidadoptima(ActionEvent e) {

ConfigurarCantidadOptima d = new ConfigurarCantidadOptima();

d.setLocationRelativeTo(mntmConfigurarcantidadoptima);

d.setVisible(true);

}

protected void actionPerformedMntmConfigurarcuotadiaria(ActionEvent e) {

ConfigurarCuotaDiaria d = new ConfigurarCuotaDiaria();

d.setLocationRelativeTo(mntmConfigurarcuotadiaria);

d.setVisible(true);

}

protected void actionPerformedMntmAcerdadetienda(ActionEvent e) {

AcercaDeTienda d = new AcercaDeTienda();

d.setLocationRelativeTo(mntmAcerdadetienda);

d.setVisible(true);

}

}

package cibertec;

import java.awt.BorderLayout;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JTextArea;

import javax.swing.JScrollPane;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.Font;

public class ListarCerámicos extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private JButton btnListar;

private JButton btnCerrar;

private JScrollPane scp;

private JTextArea txtS;

String modeloSeleccionado;

void imprimir(String cad) {

txtS.append(cad + "");

}

public static void main(String[] args) {

try {

ListarCerámicos dialog = new ListarCerámicos();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ListarCerámicos() {

setTitle("Listado de Cerámicos");

setBounds(100, 100, 624, 503);

getContentPane().setLayout(new BorderLayout());

contentPanel.setBorder(new EmptyBorder(5, 5, 5, 5));

getContentPane().add(contentPanel, BorderLayout.CENTER);

contentPanel.setLayout(null);

btnListar = new JButton("Listar");

btnListar.addActionListener(this);

btnListar.setBounds(208, 438, 94, 23);

btnListar.setFont(new Font("arial", Font.PLAIN, 12));

contentPanel.add(btnListar);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setBounds(312, 438, 84, 23);

btnCerrar.setFont(new Font("arial", Font.PLAIN, 12));

contentPanel.add(btnCerrar);

scp = new JScrollPane();

scp.setBounds(10, 10, 590, 418);

contentPanel.add(scp);

txtS = new JTextArea();

txtS.setFont(new Font("arial", Font.PLAIN, 14));

scp.setViewportView(txtS);

}

void imprimirCeramico(String modelo,double precio, double ancho, double largo, double espesor, int contenido) {

txtS.append("\n");

txtS.append("Modelo \t: " + modelo + "\n");

txtS.append("Precio \t: " + precio + "\n");

txtS.append("Ancho(cm)\t: " + ancho + "\n");

txtS.append("Largo(cm)\t: " + largo + "\n");

txtS.append("Espesor(cm) \t: " + espesor + "\n");

txtS.append("Contenido \t: " + contenido + "\n");

}

void mostrarResultados() {

txtS.setText("LISTADO DE CERÁMICOS" + "\n");

imprimirCeramico(MenúCerámicos.modelo0, MenúCerámicos.precio0, MenúCerámicos.ancho0, MenúCerámicos.largo0, MenúCerámicos.espesor0 , MenúCerámicos.contenido0);

imprimirCeramico(MenúCerámicos.modelo1, MenúCerámicos.precio1, MenúCerámicos.ancho1, MenúCerámicos.largo1, MenúCerámicos.espesor1 , MenúCerámicos.contenido1);

imprimirCeramico(MenúCerámicos.modelo2, MenúCerámicos.precio2, MenúCerámicos.ancho2, MenúCerámicos.largo2, MenúCerámicos.espesor2 , MenúCerámicos.contenido2);

imprimirCeramico(MenúCerámicos.modelo3, MenúCerámicos.precio3, MenúCerámicos.ancho3, MenúCerámicos.largo3, MenúCerámicos.espesor3 , MenúCerámicos.contenido3);

imprimirCeramico(MenúCerámicos.modelo4, MenúCerámicos.precio4, MenúCerámicos.ancho4, MenúCerámicos.largo4, MenúCerámicos.espesor4 , MenúCerámicos.contenido4);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnCerrar) {

actionPerformedBtnCerrar(e);

}

if (e.getSource() == btnListar) {

actionPerformedBtnListar(e);

}

}

protected void actionPerformedBtnListar(ActionEvent e) {

mostrarResultados(); // llamando al método mostrarResultados();

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

dispose();

}

}

package cibertec;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.JLabel;

import java.awt.Font;

import java.awt.event.ActionEvent;

import javax.swing.JComboBox;

import javax.swing.JScrollPane;

import javax.swing.JTextArea;

import javax.swing.DefaultComboBoxModel;

import java.awt.event.ActionListener;

import java.text.DecimalFormat;

public class GenerarReportes extends JDialog implements ActionListener {

// Formateador

DecimalFormat df = new DecimalFormat("0.00");

private final JPanel contentPanel = new JPanel();

private static JTextArea txtS;

private JButton btnCerrar\_1;

private JComboBox cboTipoDeReporte\_1;

public static void main(String[] args) {

try {

GenerarReportes dialog = new GenerarReportes();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public GenerarReportes() {

setTitle("Generar Reportes");

setBounds(100, 100, 734, 438);

getContentPane().setLayout(null);

{

JLabel lblTipoDeReporte = new JLabel("Tipo de reporte");

lblTipoDeReporte.setFont(new Font("Garamond", Font.PLAIN, 14));

lblTipoDeReporte.setBounds(31, 23, 118, 30);

getContentPane().add(lblTipoDeReporte);

}

cboTipoDeReporte\_1 = new JComboBox();

cboTipoDeReporte\_1.addActionListener(this);

cboTipoDeReporte\_1.setModel(new DefaultComboBoxModel(

new String[] { "Ventas por modelo", "Comparación de precios con el precio promedio",

"Comparación de cajas vendidas con la cantidad óptima ", "Estadística sobre el precio" }));

cboTipoDeReporte\_1.setFont(new Font("arial", Font.PLAIN, 14));

cboTipoDeReporte\_1.setBounds(159, 26, 255, 27);

getContentPane().add(cboTipoDeReporte\_1);

btnCerrar\_1 = new JButton("Cerrar");

btnCerrar\_1.addActionListener(this);

btnCerrar\_1.setFont(new Font("arial", Font.PLAIN, 14));

btnCerrar\_1.setBounds(586, 26, 94, 27);

getContentPane().add(btnCerrar\_1);

JScrollPane scp = new JScrollPane();

scp.setBounds(10, 93, 700, 298);

getContentPane().add(scp);

txtS = new JTextArea();

txtS.setFont(new Font("arial", Font.PLAIN, 14));

scp.setViewportView(txtS);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == cboTipoDeReporte\_1) {

actionPerformedCboTipoDeReporte(e);

}

if (e.getSource() == btnCerrar\_1) {

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

dispose(); // no cierra la aplicación en sí misma, solo libera los recursos asociados con la ventana específica.

}

/\*

\*

public static String modelo0 = "Cinza Plus";

public static double precio0 = 92.56;

public static double ancho0 = 62.0;

public static double largo0 = 62.0;

public static double espesor0 = 8;

public static int contenido0 = 6;

\*

\* \*/

private void VentasPorModelo(String modelo, int CantidadVentas, int CantidadCajasVendidas, String ImporteTotalVendido, String AporteACuotaDiaria) {

txtS.append("\nModelo \t\t\t:\t " + modelo);

txtS.append("\nCantidad de ventas \t\t:\t " + CantidadVentas);

txtS.append("\nCantidad de cajas Vendidas \t:\t " + CantidadCajasVendidas);

txtS.append("\nImporte total vendiodo \t\t:\t " + ImporteTotalVendido);

txtS.append("\nAporte a la cuota diaria \t\t:\t " + AporteACuotaDiaria + "\n\n");

}

/\*

Object

├── Number

│ ├── Integer

│ ├── Double

│ └── ...

├── String

├── Boolean

├── Character

├── Array

├── ...

└── (otras clases del usuario)

\*

\*/

private void ComparacionPrecioProm(String modelo, String Precio, String PrecioProm, Object object) {

txtS.append("\n\nModelo \t\t\t: " + modelo);

txtS.append("\nPrecio \t\t\t: " + Precio);

txtS.append("\nPrecio Promedio \t\t: " + PrecioProm);

txtS.append("\nComparación \t\t: " + object );

}

private void ComparacionCajasVendidas(String modelo, int cantidadCajasVendidas0, int cantidadOptima, Object object) {

txtS.append("\n\nModelo \t\t\t\t: " + modelo);

txtS.append("\nCantidad de cajas vendidas \t\t: " + cantidadCajasVendidas0);

txtS.append("\nCantidad óptima \t\t\t: " + cantidadOptima);

txtS.append("\nComparación \t\t\t: " + object);

}

protected void actionPerformedCboTipoDeReporte(ActionEvent e) {

switch (cboTipoDeReporte\_1.getSelectedIndex()) {

case 0:

txtS.setText("Ventas por Modelo\n");

VentasPorModelo(MenúCerámicos.modelo0, MenúCerámicos.cantidadVentas0, MenúCerámicos.cantidadCajasVendidas0, df.format(MenúCerámicos.importeTotalVendido0), df.format(MenúCerámicos.aporteACuotaDiaria0));

VentasPorModelo(MenúCerámicos.modelo1, MenúCerámicos.cantidadVentas1, MenúCerámicos.cantidadCajasVendidas1, df.format(MenúCerámicos.importeTotalVendido1), df.format(MenúCerámicos.aporteACuotaDiaria1));

VentasPorModelo(MenúCerámicos.modelo2, MenúCerámicos.cantidadVentas2, MenúCerámicos.cantidadCajasVendidas2, df.format(MenúCerámicos.importeTotalVendido2), df.format(MenúCerámicos.aporteACuotaDiaria2));

VentasPorModelo(MenúCerámicos.modelo3, MenúCerámicos.cantidadVentas3, MenúCerámicos.cantidadCajasVendidas3, df.format(MenúCerámicos.importeTotalVendido3), df.format(MenúCerámicos.aporteACuotaDiaria3));

VentasPorModelo(MenúCerámicos.modelo4, MenúCerámicos.cantidadVentas4, MenúCerámicos.cantidadCajasVendidas4, df.format(MenúCerámicos.importeTotalVendido4), df.format(MenúCerámicos.aporteACuotaDiaria4));

break;

case 1:

txtS.setText("COMPARACIÓN DE PRECIOS CON EL PRECIO PROMEDIO");

ComparacionPrecioProm(MenúCerámicos.modelo0, df.format(MenúCerámicos.precio0), df.format(MenúCerámicos.importeTotalVendido0 / MenúCerámicos.cantidadCajasVendidas0), imprimir(MenúCerámicos.precio0, (MenúCerámicos.importeTotalVendido0 / MenúCerámicos.cantidadCajasVendidas0), 1));

ComparacionPrecioProm(MenúCerámicos.modelo1, df.format(MenúCerámicos.precio1), df.format(MenúCerámicos.importeTotalVendido1 / MenúCerámicos.cantidadCajasVendidas1), imprimir(MenúCerámicos.precio1, (MenúCerámicos.importeTotalVendido1 / MenúCerámicos.cantidadCajasVendidas1), 1));

ComparacionPrecioProm(MenúCerámicos.modelo2, df.format(MenúCerámicos.precio2), df.format(MenúCerámicos.importeTotalVendido2 / MenúCerámicos.cantidadCajasVendidas2), imprimir(MenúCerámicos.precio2, (MenúCerámicos.importeTotalVendido2 / MenúCerámicos.cantidadCajasVendidas2), 1));

ComparacionPrecioProm(MenúCerámicos.modelo3, df.format(MenúCerámicos.precio3), df.format(MenúCerámicos.importeTotalVendido3 / MenúCerámicos.cantidadCajasVendidas3), imprimir(MenúCerámicos.precio3, (MenúCerámicos.importeTotalVendido3 / MenúCerámicos.cantidadCajasVendidas2), 1));

ComparacionPrecioProm(MenúCerámicos.modelo4, df.format(MenúCerámicos.precio4), df.format(MenúCerámicos.importeTotalVendido4 / MenúCerámicos.cantidadCajasVendidas4), imprimir(MenúCerámicos.precio4, (MenúCerámicos.importeTotalVendido3 / MenúCerámicos.cantidadCajasVendidas2), 1));

break;

case 2:

txtS.setText("COMPARACIÓN DE CAJAS VENDIDAS CON LA CANTIDAD ÓPTIMA");

ComparacionCajasVendidas(MenúCerámicos.modelo0, MenúCerámicos.cantidadCajasVendidas0, MenúCerámicos.cantidadOptima, imprimir(MenúCerámicos.cantidadCajasVendidas0, MenúCerámicos.cantidadOptima, 2));

ComparacionCajasVendidas(MenúCerámicos.modelo1, MenúCerámicos.cantidadCajasVendidas1, MenúCerámicos.cantidadOptima, imprimir(MenúCerámicos.cantidadCajasVendidas1, MenúCerámicos.cantidadOptima, 2));

ComparacionCajasVendidas(MenúCerámicos.modelo2, MenúCerámicos.cantidadCajasVendidas2, MenúCerámicos.cantidadOptima, imprimir(MenúCerámicos.cantidadCajasVendidas2, MenúCerámicos.cantidadOptima, 2));

ComparacionCajasVendidas(MenúCerámicos.modelo3, MenúCerámicos.cantidadCajasVendidas3, MenúCerámicos.cantidadOptima, imprimir(MenúCerámicos.cantidadCajasVendidas3, MenúCerámicos.cantidadOptima, 2));

ComparacionCajasVendidas(MenúCerámicos.modelo4, MenúCerámicos.cantidadCajasVendidas4, MenúCerámicos.cantidadOptima, imprimir(MenúCerámicos.cantidadCajasVendidas4, MenúCerámicos.cantidadOptima, 2));

break;

case 3:

txtS.setText("ESTADISTICA SOBRE EL PRECIO\n\n");

txtS.append("\nPrecio Prom\t: S/." +(MenúCerámicos.precioProm));

txtS.append("\nPrecio Mayor\t: S/." +(MenúCerámicos.precio0));

txtS.append("\nPrecio Menor\t: S/"+(MenúCerámicos.precio1));

break;

default:

break;

}

}

/\*

\* void => el método realiza alguna acción o tarea(del código), pero no produce un resultado que pueda ser utilizado después(como un return).

\* Object => estás indicando que el método puede devolver cualquier objeto,

\*

\*/

Object imprimir(double cantidad, double valor, int bandera) {

String msg = " la cantidad óptima\n";

if (bandera == 1) {

msg = " el promedio\n";

}

if (valor > cantidad) {

return df.format(valor - cantidad) + " menos que" + msg;

} else if (valor < cantidad) {

return df.format(cantidad - valor) + " más que" + msg;

} else if (valor == cantidad) {

return "igual que" + msg;

} else {

return"\n";

}

}

}

package cibertec;

import java.awt.FlowLayout;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JTextField;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import javax.swing.DefaultComboBoxModel;

import java.awt.Font;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class ConsultarCerámico extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private static JTextField txtPrecio;

private static JTextField txtAncho;

private static JTextField txtLargo;

private static JLabel lblEspesor;

private JTextField txtEspesor;

private JTextField txtContenido;

String modeloSeleccionado;

private JComboBox cboModelo;

private JButton btnCerrar;

public static void main(String[] args) {

try {

ConsultarCerámico dialog = new ConsultarCerámico();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ConsultarCerámico() {

setTitle("Consultar Cerámico");

setBounds(100, 100, 500, 261);

getContentPane().setLayout(null);

contentPanel.setBounds(0, 0, 0, 0);

contentPanel.setLayout(new FlowLayout());

contentPanel.setBorder(new EmptyBorder(5, 5, 5, 5));

getContentPane().add(contentPanel);

JLabel lblModelo = new JLabel("Modelo ");

lblModelo.setFont(new Font("arial", Font.PLAIN, 12));

lblModelo.setBounds(10, 26, 68, 21);

getContentPane().add(lblModelo);

cboModelo = new JComboBox();

cboModelo.addActionListener(this);

cboModelo.setFont(new Font("arial", Font.PLAIN, 12));

cboModelo.setModel(new DefaultComboBoxModel(new String[]{"Cinza Plus", "Luxury", "Austria", "Yungay Mix", "Thalía"}));

cboModelo.setBounds(129, 23, 220, 21);

getContentPane().add(cboModelo);

txtPrecio = new JTextField();

txtPrecio.setFont(new Font("arial", Font.PLAIN, 12));

txtPrecio.setEditable(false);

txtPrecio.setBounds(129, 56, 220, 19);

getContentPane().add(txtPrecio);

txtPrecio.setColumns(10);

JLabel lblPrecio = new JLabel("Precio (S/)");

lblPrecio.setFont(new Font("arial", Font.PLAIN, 12));

lblPrecio.setBounds(10, 57, 89, 21);

getContentPane().add(lblPrecio);

JLabel lblContenido = new JLabel("Contenido");

lblContenido.setFont(new Font("arial", Font.PLAIN, 12));

lblContenido.setBounds(10, 181, 68, 21);

getContentPane().add(lblContenido);

JLabel lblAncho = new JLabel("Ancho (cm)");

lblAncho.setFont(new Font("arial", Font.PLAIN, 12));

lblAncho.setBounds(10, 88, 89, 21);

getContentPane().add(lblAncho);

JLabel lblLargo = new JLabel("Largo (cm)");

lblLargo.setFont(new Font("arial", Font.PLAIN, 12));

lblLargo.setBounds(10, 119, 89, 21);

getContentPane().add(lblLargo);

lblEspesor = new JLabel("Espesor (mm)");

lblEspesor.setFont(new Font("arial", Font.PLAIN, 12));

lblEspesor.setBounds(10, 150, 109, 21);

getContentPane().add(lblEspesor);

txtAncho = new JTextField();

txtAncho.setFont(new Font("arial", Font.PLAIN, 12));

txtAncho.setEditable(false);

txtAncho.setColumns(10);

txtAncho.setBounds(129, 87, 220, 19);

getContentPane().add(txtAncho);

txtLargo = new JTextField();

txtLargo.setFont(new Font("arial", Font.PLAIN, 12));

txtLargo.setEditable(false);

txtLargo.setColumns(10);

txtLargo.setBounds(129, 118, 220, 19);

getContentPane().add(txtLargo);

txtEspesor = new JTextField();

txtEspesor.setFont(new Font("arial", Font.PLAIN, 12));

txtEspesor.setEditable(false);

txtEspesor.setColumns(10);

txtEspesor.setBounds(129, 151, 220, 19);

getContentPane().add(txtEspesor);

txtContenido = new JTextField();

txtContenido.setFont(new Font("arial", Font.PLAIN, 12));

txtContenido.setEditable(false);

txtContenido.setColumns(10);

txtContenido.setBounds(129, 182, 220, 19);

getContentPane().add(txtContenido);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setFont(new Font("arial", Font.PLAIN, 12));

btnCerrar.setBounds(391, 24, 85, 21);

getContentPane().add(btnCerrar);

txtPrecio.setText(MenúCerámicos.precio0 + "");

txtAncho.setText(MenúCerámicos.ancho0 + "");

txtLargo.setText(MenúCerámicos.largo0 + "");

txtEspesor.setText(MenúCerámicos.espesor0 + "");

txtContenido.setText(MenúCerámicos.contenido0 + "");

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnCerrar) {

actionPerformedBtnCerrar(e);

}

if (e.getSource() == cboModelo) {

actionPerformedCboModelo(e);

}

}

void imprimirDatosDeCeramica(double precio, double ancho, double largo, double espesor, int contenido) {

txtPrecio.setText(precio + "");

txtAncho.setText(ancho + "");

txtLargo.setText(largo + "");

txtEspesor.setText(espesor + "");

txtContenido.setText(contenido + "");

}

protected void actionPerformedCboModelo(ActionEvent e) {

switch (cboModelo.getSelectedIndex()) {

case 0:

imprimirDatosDeCeramica(MenúCerámicos.precio0, MenúCerámicos.ancho0, MenúCerámicos.largo0, MenúCerámicos.espesor0, MenúCerámicos.contenido0);

modeloSeleccionado = "0";

break;

case 1:

imprimirDatosDeCeramica(MenúCerámicos.precio1, MenúCerámicos.ancho1, MenúCerámicos.largo1, MenúCerámicos.espesor1, MenúCerámicos.contenido1);

modeloSeleccionado = "1";

break;

case 2:

imprimirDatosDeCeramica(MenúCerámicos.precio2, MenúCerámicos.ancho2, MenúCerámicos.largo2, MenúCerámicos.espesor2, MenúCerámicos.contenido2);

modeloSeleccionado = "2";

break;

case 3:

imprimirDatosDeCeramica(MenúCerámicos.precio3, MenúCerámicos.ancho3, MenúCerámicos.largo3, MenúCerámicos.espesor3, MenúCerámicos.contenido3);

modeloSeleccionado = "3";

break;

default:

imprimirDatosDeCeramica(MenúCerámicos.precio4, MenúCerámicos.ancho4, MenúCerámicos.largo4, MenúCerámicos.espesor4, MenúCerámicos.contenido0);

modeloSeleccionado = "4";

break;

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

dispose();

}

}

package cibertec;

import java.awt.BorderLayout;

import java.awt.FlowLayout;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import java.awt.Font;

import javax.swing.JLabel;

import javax.swing.JTextField;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class ConfigurarObsequios extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private static JTextField txtTipodeobsequio;

private static JTextField txt1a5;

private static JTextField txt6a10;

private static JTextField txt11amas;

private JButton btnCancelar;

private JButton btnAceptar;

public static void main(String[] args) {

try {

ConfigurarObsequios dialog = new ConfigurarObsequios();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ConfigurarObsequios() {

setFont(new Font("Dialog", Font.PLAIN, 14));

setTitle("Configurar obsequios");

setBounds(100, 100, 450, 180);

getContentPane().setLayout(null);

{

JLabel lblTipodeobsequio = new JLabel("Tipo de obsequio");

lblTipodeobsequio.setFont(new Font("arial", Font.PLAIN, 12));

lblTipodeobsequio.setBounds(10, 20, 130, 31);

getContentPane().add(lblTipodeobsequio);

}

{

JLabel lbl1a5unidades = new JLabel("1 a 5 unidades");

lbl1a5unidades.setFont(new Font("arial", Font.PLAIN, 12));

lbl1a5unidades.setBounds(10, 45, 130, 31);

getContentPane().add(lbl1a5unidades);

}

{

JLabel lbl6a10unidades = new JLabel("6 a 10 unidades");

lbl6a10unidades.setFont(new Font("arial", Font.PLAIN, 12));

lbl6a10unidades.setBounds(10, 74, 130, 31);

getContentPane().add(lbl6a10unidades);

}

{

JLabel lbl11amasunidades = new JLabel("11 a más unidades");

lbl11amasunidades.setFont(new Font("arial", Font.PLAIN, 12));

lbl11amasunidades.setBounds(10, 102, 130, 31);

getContentPane().add(lbl11amasunidades);

}

{

txtTipodeobsequio = new JTextField();

txtTipodeobsequio.setFont(new Font("arial", Font.PLAIN, 12));

txtTipodeobsequio.setBounds(153, 24, 160, 19);

getContentPane().add(txtTipodeobsequio);

txtTipodeobsequio.setColumns(10);

}

{

txt1a5 = new JTextField();

txt1a5.setFont(new Font("arial", Font.PLAIN, 12));

txt1a5.setColumns(10);

txt1a5.setBounds(153, 49, 160, 19);

getContentPane().add(txt1a5);

}

{

txt6a10 = new JTextField();

txt6a10.setFont(new Font("arial", Font.PLAIN, 12));

txt6a10.setColumns(10);

txt6a10.setBounds(153, 78, 160, 19);

getContentPane().add(txt6a10);

}

{

txt11amas = new JTextField();

txt11amas.setFont(new Font("arial", Font.PLAIN, 12));

txt11amas.setColumns(10);

txt11amas.setBounds(153, 106, 160, 19);

getContentPane().add(txt11amas);

}

txtTipodeobsequio.setText(MenúCerámicos.tipoObsequio + "");

txt1a5.setText(MenúCerámicos.obsequioCantidad1 + "");

txt6a10.setText(MenúCerámicos.obsequioCantidad2 + "");

txt11amas.setText(MenúCerámicos.obsequioCantidad3 + "");

{

btnCancelar = new JButton("Cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setFont(new Font("arial", Font.PLAIN, 12));

btnCancelar.setBounds(323, 48, 103, 24);

getContentPane().add(btnCancelar);

}

{

btnAceptar = new JButton("Aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setFont(new Font("arial", Font.PLAIN, 12));

btnAceptar.setBounds(323, 23, 103, 24);

getContentPane().add(btnAceptar);

}

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnAceptar) {

actionPerformedBtnAceptar(e);

}

if (e.getSource() == btnCancelar) {

actionPerformedBtnCancelar(e);

}

}

protected void actionPerformedBtnCancelar(ActionEvent e) {

dispose();

}

protected void actionPerformedBtnAceptar(ActionEvent e) {

MenúCerámicos.tipoObsequio = String.valueOf(txtTipodeobsequio.getText());

MenúCerámicos.obsequioCantidad1 = Integer.parseInt(txt1a5.getText());

MenúCerámicos.obsequioCantidad2 = Integer.parseInt(txt6a10.getText());

MenúCerámicos.obsequioCantidad3 = Integer.parseInt(txt11amas.getText());

dispose();

}

}

package cibertec;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.JLabel;

import javax.swing.JTextField;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.Font;

public class ConfigurarDescuentos extends JDialog implements ActionListener {

private static JTextField txtmasde15;

private static JTextField txt11a15;

private static JTextField txt6a10;

private static JTextField txt1a5;

private final JPanel contentPanel = new JPanel();

private JButton btnAceptar;

private JButton btnCancelar;

public static void main(String[] args) {

try {

ConfigurarDescuentos dialog = new ConfigurarDescuentos();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ConfigurarDescuentos() {

setTitle("Configurar porcentajes de descuento");

setBounds(100, 100, 463, 173);

getContentPane().setLayout(null);

JLabel lbl1a5 = new JLabel("1 a 5 unidades");

lbl1a5.setFont(new Font("arial", Font.PLAIN, 12));

lbl1a5.setBounds(10, 15, 119, 14);

getContentPane().add(lbl1a5);

JLabel lbl6a10 = new JLabel("6 a 10 unidades");

lbl6a10.setFont(new Font("arial", Font.PLAIN, 12));

lbl6a10.setBounds(10, 44, 119, 14);

getContentPane().add(lbl6a10);

JLabel lbl11a15 = new JLabel("11 a 15 unidades");

lbl11a15.setFont(new Font("arial", Font.PLAIN, 12));

lbl11a15.setBounds(10, 73, 119, 14);

getContentPane().add(lbl11a15);

JLabel lblmasde15 = new JLabel("Más de 15 unidades");

lblmasde15.setFont(new Font("Garamond", Font.PLAIN, 14));

lblmasde15.setBounds(10, 102, 119, 14);

getContentPane().add(lblmasde15);

txtmasde15 = new JTextField();

txtmasde15.setFont(new Font("arial", Font.PLAIN, 12));

txtmasde15.setBounds(128, 100, 86, 20);

getContentPane().add(txtmasde15);

txtmasde15.setColumns(10);

txt11a15 = new JTextField();

txt11a15.setFont(new Font("arial", Font.PLAIN, 12));

txt11a15.setColumns(10);

txt11a15.setBounds(128, 70, 86, 20);

getContentPane().add(txt11a15);

txt6a10 = new JTextField();

txt6a10.setFont(new Font("arial", Font.PLAIN, 12));

txt6a10.setColumns(10);

txt6a10.setBounds(128, 40, 86, 20);

getContentPane().add(txt6a10);

txt1a5 = new JTextField();

txt1a5.setFont(new Font("arial", Font.PLAIN, 12));

txt1a5.setColumns(10);

txt1a5.setBounds(128, 10, 86, 20);

getContentPane().add(txt1a5);

JLabel lblPorcen1a5 = new JLabel("%");

lblPorcen1a5.setFont(new Font("arial", Font.PLAIN, 12));

lblPorcen1a5.setBounds(224, 15, 46, 14);

getContentPane().add(lblPorcen1a5);

JLabel lblPorcen6a10 = new JLabel("%");

lblPorcen6a10.setFont(new Font("arial", Font.PLAIN, 12));

lblPorcen6a10.setBounds(224, 44, 46, 14);

getContentPane().add(lblPorcen6a10);

JLabel lblPorcen11a15 = new JLabel("%");

lblPorcen11a15.setFont(new Font("arial", Font.PLAIN, 12));

lblPorcen11a15.setBounds(224, 73, 46, 14);

getContentPane().add(lblPorcen11a15);

JLabel lblPorcenmasde15 = new JLabel("%");

lblPorcenmasde15.setFont(new Font("arial", Font.PLAIN, 12));

lblPorcenmasde15.setBounds(224, 102, 46, 14);

getContentPane().add(lblPorcenmasde15);

btnAceptar = new JButton("Aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setFont(new Font("arial", Font.PLAIN, 12));

btnAceptar.setBounds(315, 11, 113, 23);

getContentPane().add(btnAceptar);

txt1a5.setText(MenúCerámicos.porcentaje1 + "");

txt6a10.setText(MenúCerámicos.porcentaje2 + "");

txt11a15.setText(MenúCerámicos.porcentaje3 + "");

txtmasde15.setText(MenúCerámicos.porcentaje4 + "");

btnCancelar = new JButton("Cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setFont(new Font("arial", Font.PLAIN, 12));

btnCancelar.setBounds(315, 40, 113, 23);

getContentPane().add(btnCancelar);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnCancelar) {

actionPerformedBtnCancelar(e);

}

if (e.getSource() == btnAceptar) {

actionPerformedBtnAceptar(e);

}

}

protected void actionPerformedBtnAceptar(ActionEvent e) {

MenúCerámicos.porcentaje1 = Double.parseDouble(txt1a5.getText());

MenúCerámicos.porcentaje2 = Double.parseDouble(txt6a10.getText());

MenúCerámicos.porcentaje3 = Double.parseDouble(txt11a15.getText());

MenúCerámicos.porcentaje4 = Double.parseDouble(txtmasde15.getText());

dispose();

}

protected void actionPerformedBtnCancelar(ActionEvent e) {

dispose();

}

}

package cibertec;

import java.awt.BorderLayout;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;

import javax.swing.JTextField;

import java.awt.Font;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class ConfigurarCuotaDiaria extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private JLabel lblCuota;

private static JTextField txtCuotaD;

private JButton btnAceptar;

private JButton btnCancelar;

public static void main(String[] args) {

try {

ConfigurarCuotaDiaria dialog = new ConfigurarCuotaDiaria();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ConfigurarCuotaDiaria() {

setTitle("Configurar cuota diaria");

setBounds(100, 100, 484, 141);

getContentPane().setLayout(new BorderLayout());

contentPanel.setBorder(new EmptyBorder(5, 5, 5, 5));

getContentPane().add(contentPanel, BorderLayout.CENTER);

contentPanel.setLayout(null);

lblCuota = new JLabel("Cuota diaria esperada (S/.)");

lblCuota.setFont(new Font("Arial", Font.PLAIN, 14));

lblCuota.setBounds(21, 22, 187, 20);

contentPanel.add(lblCuota);

txtCuotaD = new JTextField();

txtCuotaD.setFont(new Font("Arial", Font.PLAIN, 12));

txtCuotaD.setBounds(213, 18, 108, 30);

contentPanel.add(txtCuotaD);

txtCuotaD.setColumns(10);

btnAceptar = new JButton("Aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setFont(new Font("Arial", Font.PLAIN, 12));

btnAceptar.setBounds(357, 18, 89, 30);

contentPanel.add(btnAceptar);

txtCuotaD.setText(MenúCerámicos.cuotaDiaria + "");

btnCancelar = new JButton("Cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setFont(new Font("Arial", Font.PLAIN, 12));

btnCancelar.setBounds(357, 56, 89, 30);

contentPanel.add(btnCancelar);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnAceptar) {

actionPerformedBtnAceptar(e);

}

if (e.getSource() == btnCancelar) {

actionPerformedBtnCancelar(e);

}

}

protected void actionPerformedBtnCancelar(ActionEvent e) {

dispose();

}

protected void actionPerformedBtnAceptar(ActionEvent e) {

MenúCerámicos.cuotaDiaria = Double.parseDouble(txtCuotaD.getText());

}

}

package cibertec;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextField;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class ConfigurarCantidadOptima extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private static JTextField txtCantidadOptima;

private JButton btnCancelar;

private JButton btnAceptar;

public static void main(String[] args) {

try {

ConfigurarCantidadOptima dialog = new ConfigurarCantidadOptima();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public ConfigurarCantidadOptima() {

setTitle("Configurar cantidad óptima");

setBounds(100, 100, 401, 142);

getContentPane().setLayout(null);

JLabel lblCantidadOptima = new JLabel("Cantidad óptima de unidades vendidas");

lblCantidadOptima.setBounds(10, 16, 247, 38);

lblCantidadOptima.setFont(new Font("Arial", Font.PLAIN, 13));

getContentPane().add(lblCantidadOptima);

txtCantidadOptima = new JTextField();

txtCantidadOptima.setBounds(31, 57, 196, 25);

txtCantidadOptima.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

}

});

txtCantidadOptima.setFont(new Font("Monospaced", Font.PLAIN, 12));

getContentPane().add(txtCantidadOptima);

txtCantidadOptima.setColumns(10);

txtCantidadOptima.setText(MenúCerámicos.cantidadOptima + "");

btnAceptar = new JButton("Aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setBounds(267, 20, 101, 30);

btnAceptar.setFont(new Font("Arial", Font.PLAIN, 11));

getContentPane().add(btnAceptar);

btnCancelar = new JButton("Cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setBounds(267, 55, 101, 30);

btnCancelar.setFont(new Font("Arial", Font.PLAIN, 11));

getContentPane().add(btnCancelar);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnAceptar) {

actionPerformedBtnAceptar(e);

}

if (e.getSource() == btnCancelar) {

actionPerformedBtnCancelar(e);

}

}

protected void actionPerformedBtnCancelar(ActionEvent e) {

dispose();

}

protected void actionPerformedBtnAceptar(ActionEvent e) {

MenúCerámicos.cantidadOptima = Integer.parseInt(txtCantidadOptima.getText());

}

}

package cibertec;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JPanel;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.JSeparator;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.SwingConstants;

public class AcercaDeTienda extends JDialog implements ActionListener {

private final JPanel contentPanel = new JPanel();

private JButton btnCerrar;

public static void main(String[] args) {

try {

AcercaDeTienda dialog = new AcercaDeTienda();

dialog.setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

public AcercaDeTienda() {

setTitle("Acerca de Tienda");

setBounds(100, 100, 475, 335);

getContentPane().setLayout(null);

JLabel lblJesusChavez = new JLabel("Johnny Nestor Jair Ramos Paucar (i202334672) ");

lblJesusChavez.setFont(new Font("Garamond", Font.PLAIN, 16));

lblJesusChavez.setBounds(77, 91, 304, 26);

getContentPane().add(lblJesusChavez);

JLabel lblAutores = new JLabel("Autores");

lblAutores.setFont(new Font("Garamond", Font.PLAIN, 27));

lblAutores.setBounds(188, 14, 83, 63);

getContentPane().add(lblAutores);

JLabel lblBetsyPantaleon = new JLabel("Laysson Jamir Polo De La Cruz (i202225224)");

lblBetsyPantaleon.setFont(new Font("Garamond", Font.PLAIN, 16));

lblBetsyPantaleon.setBounds(85, 131, 288, 26);

getContentPane().add(lblBetsyPantaleon);

JLabel lblJuniorRodriguez = new JLabel("Mijael Jefferson Esquivel Almanza");

lblJuniorRodriguez.setFont(new Font("Garamond", Font.PLAIN, 16));

lblJuniorRodriguez.setBounds(117, 171, 224, 26);

getContentPane().add(lblJuniorRodriguez);

JLabel lblMelannyAtoche = new JLabel("Juan Sebastian Guzman Ramos");

lblMelannyAtoche.setFont(new Font("Garamond", Font.PLAIN, 16));

lblMelannyAtoche.setBounds(131, 211, 197, 26);

getContentPane().add(lblMelannyAtoche);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setFont(new Font("Garamond", Font.PLAIN, 16));

btnCerrar.setBounds(184, 251, 91, 26);

getContentPane().add(btnCerrar);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnCerrar) {

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

dispose();

}

}

#### Conclusiones

Este proyecto de desarrollo de software para la venta de cerámicas tiene como principal mejorar la interacción con los clientes y aumentar la eficiencia de la empresa. Facilita la actualización constante del catálogo, así como la identificación de los productos más populares y aquellos que necesitan ser reemplazados.

**Mejora en la Experiencia del Cliente**: La implementación del programa de Tienda de Cerámicos tiene un impacto directo en la experiencia del cliente. Al facilitar la selección y compra de productos, así como mejorar la eficiencia en la atención al cliente, se espera que los clientes perciban un servicio de mayor calidad.

**Incremento de Eficiencia Operativa**: El proyecto tiene una pertinencia destacada al aumentar la eficiencia operativa tanto para los trabajadores de la tienda como para los proveedores de cerámicos.

**Incentivo para el Desarrollo Tecnológico:** La introducción de un programa de Tienda de Cerámicos impulsa el desarrollo tecnológico en el contexto elegido. Los desarrolladores del software involucrados en el proyecto se benefician directamente al participar en la creación y mejora continua del programa. Este impacto positivo no solo se limita al ámbito de la tienda de cerámicos, sino que también puede contribuir al crecimiento y la innovación en el sector tecnológico de la región.

#### Recomendaciones

**Entender las Necesidades del Usuario:** Antes de iniciar el desarrollo del proyecto, es crucial realizar una investigación exhaustiva para comprender las necesidades y expectativas de los usuarios finales, tanto clientes como empleados de la tienda. Ayudarán a garantizar que el programa aborde de manera efectiva los desafíos y requisitos específicos para la tienda de cerámicos y/o similiar.

**Capacitación y Soporte Continuo:** Invertir en programas de capacitación y proporcionar un sólido sistema de soporte técnico es esencial para el éxito continuo del proyecto. Los empleados de la tienda deben recibir entrenamiento adecuado para utilizar eficientemente el nuevo software, lo que contribuirá a una transición suave y a una adopción exitosa.

#### Bibliografía

Se realizó búsqueda de información  
en   
[stackoverflow](https://stackoverflow.com/)

Se Utilizo para la búsqueda y comprensión de códigos distintos, para la implementación en el proyecto.

[nexovirtual](https://nexovirtual.net/blog/factores-para-disenar-una-tienda-de-alfareria-online/)

Se utilizó para poder saber un poco más de información sobre productos relacionados con la cerámica.