

**INSTITUTO SUPERIOR TECNOLOGICO PRIVADO CIBERTEC**

**CARRERAS PROFESIONALES**

**INTRODUCCION A LA ALGORITMIA**

**SISTEMA DE VENTA DE LADRILLOS**

**PRIMER CICLO**

**SECCIÓN T1IN**

**SEMESTRE 2023 – 01**

**Coordinador:**

**JESÚS FLORES SARMIENTO**

**Integrantes:**

**JUAN DAVID ANAYA RODRIGUEZ**

**MAURO ARANA ACUÑA**

**ANDRE PEREYRA GAONA**

**ANGEL PINTADO NIZAMA**

**STEPHANO YPARRAGUIRRE CERON**

**CARRERAS PROFESIONALES**

**CIBERTEC**

**INDEPENDENCIA, 2023**

#### Índice

1. Introducción
2. Justificación
3. Objetivos
4. Definición
   1. Inicio
   2. Mantenimiento
      1. Consultar Ladrillo
      2. Modificar Ladrillo
      3. Listar Ladrillos
   3. Ventas
      1. Vender
      2. Generar Reportes
   4. Configurar
      1. Configurar descuentos
      2. Configurar obsequios
   5. Ayuda
      1. Acerca de Tienda
5. Entregables
6. Conclusiones
7. Recomendaciones
8. Bibliografía

#### Introducción

El programa “Tienda” es un software que permite a los asesores de ventas realizar:

* Consultas: Visualizar el precio, las medidas y cantidad optima de cada modelo de ladrillo.
* Modificaciones: Cambiar el precio, medidas y cantidad optima de cada modelo de ladrillo.
* Listados: Visualizar, en un reporte, todos los datos de cada modelo de ladrillo.
* Ventas: Realizar una venta de cada modelo, muestra el precio por millar y una boleta de venta.
* Reportes: Visualizar, en un reporte, todas las ventas realizadas por cada modelo, comparar el precio de los productos entre sí, comparar las ventas realizadas con las cantidades optimas, visualizar una estadística general.
* Configuraciones: Modificar los distintos descuentos que se ofrecen por las cantidades vendidas. Así como el obsequio a brindar.

#### Justificación

El programa permite al usuario realizar y consultar las ventas en el negocio, asimismo visualizar de manera resumida distintos reportes, brindando un modelo de ventas más eficiente y seguro.

#### Objetivos

* Reducir tiempo y costos
* Optimizar el proceso de ventas
* Brindar un mejor servicio al cliente

#### Definición

**Inicio:** Al abrir el programa este mostrará una ventana con una barra de navegación en la que podrá interactuar con las distintas funciones.

**Archivo:** Salir.

**Mantenimiento:** Consultar ladrillo, modificar ladrillo y listar ladrillos.

**Ventas:** Vender, generar reportes.

**Configuración:** Configurar descuentos, configurar obsequios.

**Ayuda:** Acerca de Tienda.

A screenshot of a computer

Description automatically generated

**4.2. Mantenimiento**

En mantenimiento el usuario podrá realizar consultas, modificaciones y un reporte de los distintos modelos de ladrillos.

**4.2.1. Consultar Ladrillo**

Esta opción despliega un cuadro de dialogo que muestra todos los datos de cada modelo.

Dentro de este se encuentran un cuadro combinado, cuadros de texto no editables y un botón.

Al seleccionar, en el cuadro combinado, el modelo deseado actualizará, en los cuadros de texto, los datos como:

* Precio (S/)
* Ancho (cm)
* Largo (cm)
* Espesor (mm)
* Cantidad Óptima (millares)

Finalmente el botón Cerrar cerrará el cuadro de dialogo.

A screenshot of a computer

Description automatically generated

**4.2.2 Modificar Ladrillo.**

Esta opción despliega un cuadro de dialogo similar al anterior, con la diferencia de que los cuadros de texto son editables y un botón adicional.

Los cuadros de texto editables permiten modificar los valores de cada modelo.

Al pulsar el botón Grabar se actualizará en el sistema los datos modificados.

A screenshot of a computer

Description automatically generated

Como método de confirmación se mostrará un nuevo cuadro de diálogo especificando los valores modificados, entre estos.

* Modelo al que se le hizo la modificación
* Precio
* Ancho
* Largo
* Espesor
* Cantidad Optima

A screenshot of a computer

Description automatically generated with medium confidence

En caso no se haya realizado ninguna modificación este cuadro simplemente no aparecerá.

**4.2.3. Listar ladrillos**

Esta opción despliega un cuadro de dialogo con un JTextArea scrolleable y 2 botones.

Al pulsar el botón Listar se actualizará el JTextArea y este mostrará un reporte con todos los modelos y sus respectivos datos.

Al pulsar el botón Cerrar se cerrará el cuadro de dialogo.

A screenshot of a computer

Description automatically generated

**4.3. Ventas**

En el menú Ventas se realizarán las ventas por modelos, además de generar distintos reportes.

**4.3.1. Vender**

Esta opción despliega un cuadro de diálogo con un cuadro combinado, dos cuadros de texto, dos botones y un área de texto no editable.

El cuadro combinado permite seleccionar el modelo a vender. Los cuadros de textos, no editable y editable respectivamente, permiten consultar el precio por millar e insertar la cantidad a vender por modelo.

Al pulsar el botón Vender se realizará una venta, se mostrará un pequeño cuadro de confirmación indicando que la venta ha sido realizada, además de el monto total recaudado. También se mostrará, en el área de texto, una boleta de venta indicando los siguientes datos:

* Modelo de ladrillo
* Precio por millar
* Cantidad adquirida
* Importe total
* Importe descuento
* Importe pagar
* Tipo de obsequio
* Unidades a obsequiar

El botón Cerrar cerrará el cuadro de dialogo.

A screenshot of a computer

Description automatically generated

A screenshot of a computer error

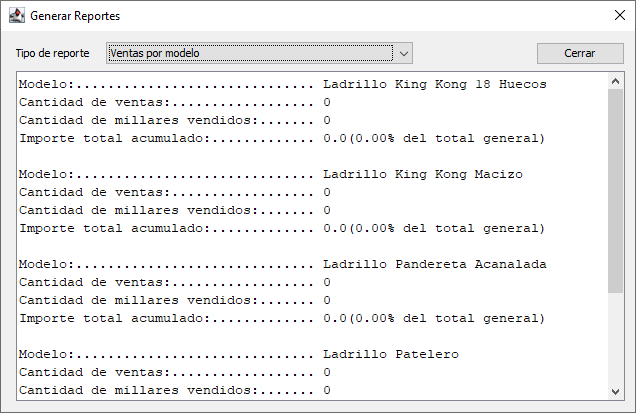
Description automatically generated with low confidence

**4.3.2 Generar Reportes**

Esta opción despliega un cuadro de diálogo con un cuadro combinado, un botón y un área de texto.

El cuadro combinado tendrá los distintos tipos de reportes, que al seleccionar actualizará el área de texto mostrando el reporte deseado, entre estos:

* Ventas por modelo
* Comparación con el precio promedio
* Comparación con la cantidad promedio
* Estadística general



A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

**4.4. Configuración:**

En el menú Configuración se harán las distintas modificaciones a los descuentos y a los obsequios, cada uno independiente a la cantidad adquirida.

**4.4.1 Configurar descuentos:**

Esta opción desplegará un cuadro de diálogo con 5 cuadros de texto y 2 botones, ahí se modificarán los distintos porcentajes dependiendo de la cantidad adquirida, entre estos encontramos los descuentos cuando se compran:

* 1 a 5 millares
* 6 a 10 millares
* 11 a 15 millares
* 16 a 20 millares
* Más de 20 millares

A screenshot of a computer

Description automatically generated with low confidence

**4.4.2. Configurar obsequios:**

Esta opción desplegará un cuadro de diálogo con 6 cuadros de texto y 2 botones, ahí se modificarán el tipo de obsequio y las distintas cantidades a obsequirar por cantidad comprada, estas modificaciones son:

* Tipo de obsequio
* 1 a 5 millares
* 6 a 10 millares
* 11 a 15 millares
* 16 a 20 millares
* Más de 20 millares

A screenshot of a computer

Description automatically generated with medium confidence

**4.5. Ayuda:**

**4.5.1. Acerca de Tienda:**

Esta opción desplegará un cuadro de diálogo con el nombre del programa y los datos de los desarrolladores

A screenshot of a computer

Description automatically generated

#### Entregables

**Tienda:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.UIManager;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.JMenu;

public class Tienda extends JFrame implements ActionListener {

private static final long serialVersionUID = 1L;

private JPanel contentPane;

private JMenuBar menuBar;

private JMenu mnArchivo;

private JMenuItem mntmSalir;

private JMenu mnMantenimiento;

private JMenuItem mntmConsultar;

private JMenuItem mntmModificar;

private JMenuItem mntmListar;

private JMenu mnVentas;

private JMenuItem mntmVender;

private JMenuItem mntmGenerar;

private JMenu mnConfiguracion;

private JMenuItem mntmDescuentos;

private JMenuItem mntmObsequios;

private JMenu mnAyuda;

private JMenuItem mntmAcerca;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

Tienda frame = new Tienda();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Declaracion de variables globales

\*/

// Ladrillo 0

public static String modelo0 = "Ladrillo King Kong 18 Huecos";

public static double precio0 = 610.0;

public static double ancho0 = 12.5;

public static double largo0 = 23.0;

public static double espesor0 = 9;

public static int cantidadOptima0 = 6;

public static int totalVentas0 = 0;

public static int totalMillares0 = 0;

public static double totalVendidos0 = 0;

// Ladrillo 1

public static String modelo1 = "Ladrillo King Kong Macizo";

public static double precio1 = 920.0;

public static double ancho1 = 12.0;

public static double largo1 = 21.0;

public static double espesor1 = 9.0;

public static int cantidadOptima1 = 6;

public static int totalVentas1 = 0;

public static int totalMillares1 = 0;

public static double totalVendidos1 = 0;

// Ladrillo 2

public static String modelo2 = "Ladrillo Pandereta Acanalada";

public static double precio2 = 510.0;

public static double ancho2 = 10.5;

public static double largo2 = 23;

public static double espesor2 = 9.0;

public static int cantidadOptima2 = 6;

public static int totalVentas2 = 0;

public static int totalMillares2 = 0;

public static double totalVendidos2 = 0;

// Ladrillo 3

public static String modelo3 = "Ladrillo Patelero";

public static double precio3 = 2200.0;

public static double ancho3 = 24.0;

public static double largo3 = 24.0;

public static double espesor3 = 3;

public static int cantidadOptima3 = 6;

public static int totalVentas3 = 0;

public static int totalMillares3 = 0;

public static double totalVendidos3 = 0;

// Ladrillo 4

public static String modelo4 = "Ladrillo Techo Hueco";

public static double precio4 = 2350.0;

public static double ancho4 = 30.0;

public static double largo4 = 30.0;

public static double espesor4 = 15.0;

public static int cantidadOptima4 = 6;

public static int totalVentas4 = 0;

public static int totalMillares4 = 0;

public static double totalVendidos4 = 0;

// Descuentos

public static double porcentaje1 = 5.0;

public static double porcentaje2 = 7.5;

public static double porcentaje3 = 10.0;

public static double porcentaje4 = 12.5;

public static double porcentaje5 = 15.0;

// Obsequios

public static String tipoObsequio = "Badilejo";

public static int obsequioCantidad1 = 1;

public static int obsequioCantidad2 = 3;

public static int obsequioCantidad3 = 5;

public static int obsequioCantidad4 = 7;

public static int obsequioCantidad5 = 9;

//ventas

public static double importeTotalAcumulado = 0;

public static int ventasTotales = 0;

/\*\*

\* Create the frame.

\*/

public Tienda() {

setTitle("Tienda 1.0");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 762, 465);

menuBar = new JMenuBar();

setJMenuBar(menuBar);

mnArchivo = new JMenu("Archivo");

menuBar.add(mnArchivo);

mntmSalir = new JMenuItem("Salir");

mntmSalir.addActionListener(this);

mntmSalir.setIcon(null);

mnArchivo.add(mntmSalir);

mnMantenimiento = new JMenu("Mantenimiento");

menuBar.add(mnMantenimiento);

mntmConsultar = new JMenuItem("Consultar ladrillo");

mntmConsultar.addActionListener(this);

mntmConsultar.setIcon(null);

mnMantenimiento.add(mntmConsultar);

mntmModificar = new JMenuItem("Modificar ladrillo");

mntmModificar.addActionListener(this);

mntmModificar.setIcon(null);

mnMantenimiento.add(mntmModificar);

mntmListar = new JMenuItem("Listar ladrillos");

mntmListar.addActionListener(this);

mntmListar.setIcon(null);

mnMantenimiento.add(mntmListar);

mnVentas = new JMenu("Ventas");

menuBar.add(mnVentas);

mntmVender = new JMenuItem("Vender");

mntmVender.addActionListener(this);

mntmVender.setIcon(null);

mnVentas.add(mntmVender);

mntmGenerar = new JMenuItem("Generar reportes");

mntmGenerar.addActionListener(this);

mntmGenerar.setIcon(null);

mnVentas.add(mntmGenerar);

mnConfiguracion = new JMenu("Configuración");

menuBar.add(mnConfiguracion);

mntmDescuentos = new JMenuItem("Configurar descuentos");

mntmDescuentos.addActionListener(this);

mntmDescuentos.setIcon(null);

mnConfiguracion.add(mntmDescuentos);

mntmObsequios = new JMenuItem("Configurar obsequios");

mntmObsequios.addActionListener(this);

mntmObsequios.setIcon(null);

mnConfiguracion.add(mntmObsequios);

mnAyuda = new JMenu("Ayuda");

menuBar.add(mnAyuda);

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

mntmAcerca.addActionListener(this);

mntmAcerca.setIcon(null);

mnAyuda.add(mntmAcerca);

contentPane = new JPanel();

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

setContentPane(contentPane);

contentPane.setLayout(null);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedMntmAcerca(e);

}

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

actionPerformedMntmObsequios(e);

}

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

actionPerformedMntmDescuentos(e);

}

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

actionPerformedMntmGenerar(e);

}

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

actionPerformedMntmVender(e);

}

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

actionPerformedMntmListar(e);

}

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

actionPerformedMntmModificar(e);

}

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

actionPerformedMntmConsultar(e);

}

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

actionPerformedMntmSalir(e);

}

}

// Archivo

protected void actionPerformedMntmSalir(ActionEvent e) {

System.exit(0);

}

// Mantenimiento

protected void actionPerformedMntmConsultar(ActionEvent e) {

// Declarar VARIABLE REFERENCIA que permita controlar la caja de diÃ¡logo

DialogoConsultar d;

// Crear la caja de dialogo en la VARIABLE REFERENCIA

d = new DialogoConsultar();

// Fijar la localizacion de la caja

d.setLocationRelativeTo(this);

// Hacer visible la caja de dialogo

d.setVisible(true);

}

protected void actionPerformedMntmModificar(ActionEvent e) {

DialogoModificar d = new DialogoModificar();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

protected void actionPerformedMntmListar(ActionEvent e) {

DialogoListar d = new DialogoListar();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

// Ventas

protected void actionPerformedMntmVender(ActionEvent e) {

DialogoVender d = new DialogoVender();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

protected void actionPerformedMntmGenerar(ActionEvent e) {

DialogoReportes d = new DialogoReportes();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

// Configuración

protected void actionPerformedMntmDescuentos(ActionEvent e) {

configDescuento d = new configDescuento();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

protected void actionPerformedMntmObsequios(ActionEvent e) {

configObsequio d = new configObsequio();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

// Ayuda

protected void actionPerformedMntmAcerca(ActionEvent e) {

DialogoAyuda d = new DialogoAyuda();

d.setLocationRelativeTo(this);

d.setVisible(true);

}

}

**Consultar Ladrillo:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.UIManager;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.DefaultComboBoxModel;

public class DialogoConsultar extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblModelo;

private JLabel lblPrecio;

private JLabel lblAncho;

private JLabel lblLargo;

private JLabel lblEspesor;

private JLabel lblCanOptima;

private JComboBox<String> cboModelo;

private JTextField txtPrecio;

private JTextField txtAncho;

private JTextField txtLargo;

private JTextField txtEspesor;

private JTextField txtCanOptima;

private JButton btnCerrar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoConsultar dialog = new DialogoConsultar();

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

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoConsultar() {

setModal(true);

setTitle("Consultar ladrillo");

setBounds(100, 100, 450, 210);

getContentPane().setLayout(null);

lblModelo = new JLabel("Modelo");

lblModelo.setBounds(15, 15, 120, 14);

getContentPane().add(lblModelo);

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

lblPrecio.setBounds(15, 40, 120, 14);

getContentPane().add(lblPrecio);

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

lblAncho.setBounds(15, 65, 120, 14);

getContentPane().add(lblAncho);

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

lblLargo.setBounds(15, 90, 120, 14);

getContentPane().add(lblLargo);

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

lblEspesor.setBounds(15, 115, 120, 14);

getContentPane().add(lblEspesor);

lblCanOptima = new JLabel("Can. Óptima (millares)");

lblCanOptima.setBounds(15, 140, 120, 14);

getContentPane().add(lblCanOptima);

cboModelo = new JComboBox<String>();

cboModelo.addActionListener(this);

cboModelo.setModel(new DefaultComboBoxModel<String>(

new String[] { "Ladrillo King Kong 18 Huecos", "Ladrillo King Kong Macizo",

"Ladrillo Pandereta Acanalada", "Ladrillo Patelero", "Ladrillo Techo Hueco" }));

cboModelo.setBounds(145, 11, 180, 22);

getContentPane().add(cboModelo);

txtPrecio = new JTextField();

txtPrecio.setEnabled(false);

txtPrecio.setBounds(145, 37, 180, 20);

getContentPane().add(txtPrecio);

txtPrecio.setColumns(10);

txtAncho = new JTextField();

txtAncho.setEnabled(false);

txtAncho.setBounds(145, 62, 180, 20);

getContentPane().add(txtAncho);

txtAncho.setColumns(10);

txtLargo = new JTextField();

txtLargo.setEnabled(false);

txtLargo.setBounds(145, 87, 180, 20);

getContentPane().add(txtLargo);

txtLargo.setColumns(10);

txtEspesor = new JTextField();

txtEspesor.setEnabled(false);

txtEspesor.setBounds(145, 112, 180, 20);

getContentPane().add(txtEspesor);

txtEspesor.setColumns(10);

txtCanOptima = new JTextField();

txtCanOptima.setEnabled(false);

txtCanOptima.setBounds(145, 137, 180, 20);

getContentPane().add(txtCanOptima);

txtCanOptima.setColumns(10);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

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

getContentPane().add(btnCerrar);

mostrarDatos(0);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedCboModelo(e);

}

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

protected void actionPerformedCboModelo(ActionEvent e) {

int mar;

mar = cboModelo.getSelectedIndex();

mostrarDatos(mar);

}

int calcularModelo() {

return cboModelo.getSelectedIndex();

}

void mostrarDatos(int mar) {

double pre, anc, lar, esp;

int can;

switch (mar) {

case 0:

pre = Tienda.precio0;

anc = Tienda.ancho0;

lar = Tienda.largo0;

esp = Tienda.espesor0;

can = Tienda.cantidadOptima0;

break;

case 1:

pre = Tienda.precio1;

anc = Tienda.ancho1;

lar = Tienda.largo1;

esp = Tienda.espesor1;

can = Tienda.cantidadOptima1;

break;

case 2:

pre = Tienda.precio2;

anc = Tienda.ancho2;

lar = Tienda.largo2;

esp = Tienda.espesor2;

can = Tienda.cantidadOptima2;

break;

case 3:

pre = Tienda.precio3;

anc = Tienda.ancho3;

lar = Tienda.largo3;

esp = Tienda.espesor3;

can = Tienda.cantidadOptima3;

break;

default:

pre = Tienda.precio4;

anc = Tienda.ancho4;

lar = Tienda.largo4;

esp = Tienda.espesor4;

can = Tienda.cantidadOptima4;

break;

}

txtPrecio.setText(pre + "");

txtAncho.setText(anc + "");

txtLargo.setText(lar + "");

txtEspesor.setText(esp + "");

txtCanOptima.setText(can + "");

}

}

**Modificar Ladrillo:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JComboBox;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.DefaultComboBoxModel;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class DialogoModificar extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblModelo;

private JLabel lblPrecio;

private JLabel lblAncho;

private JLabel lblLargo;

private JLabel lblEspesor;

private JLabel lblCanOptima;

private JComboBox<String> cboModelo;

private JTextField txtPrecio;

private JTextField txtAncho;

private JTextField txtLargo;

private JTextField txtEspesor;

private JTextField txtCanOptima;

private JButton btnCerrar;

private JButton btnGrabar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoModificar dialog = new DialogoModificar();

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

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoModificar() {

setModal(true);

setTitle("Modificar Ladrillo");

setBounds(100, 100, 450, 210);

getContentPane().setLayout(null);

lblModelo = new JLabel("Modelo");

lblModelo.setBounds(15, 15, 120, 14);

getContentPane().add(lblModelo);

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

lblPrecio.setBounds(15, 40, 120, 14);

getContentPane().add(lblPrecio);

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

lblAncho.setBounds(15, 65, 120, 14);

getContentPane().add(lblAncho);

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

lblLargo.setBounds(15, 90, 120, 14);

getContentPane().add(lblLargo);

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

lblEspesor.setBounds(15, 115, 120, 14);

getContentPane().add(lblEspesor);

lblCanOptima = new JLabel("Can. Óptima (millares)");

lblCanOptima.setBounds(15, 140, 120, 14);

getContentPane().add(lblCanOptima);

cboModelo = new JComboBox<String>();

cboModelo.addActionListener(this);

cboModelo.setModel(new DefaultComboBoxModel<String>(

new String[] { "Ladrillo King Kong 18 Huecos", "Ladrillo King Kong Macizo",

"Ladrillo Pandereta Acanalada", "Ladrillo Patelero", "Ladrillo Techo Hueco" }));

cboModelo.setBounds(145, 11, 180, 22);

getContentPane().add(cboModelo);

txtPrecio = new JTextField(calcularPrecio(0)+"");

txtPrecio.setColumns(10);

txtPrecio.setBounds(145, 37, 180, 20);

getContentPane().add(txtPrecio);

txtAncho = new JTextField(calcularAncho(0)+"");

txtAncho.setColumns(10);

txtAncho.setBounds(145, 62, 180, 20);

getContentPane().add(txtAncho);

txtLargo = new JTextField(calcularLargo(0) + "");

txtLargo.setColumns(10);

txtLargo.setBounds(145, 87, 180, 20);

getContentPane().add(txtLargo);

txtEspesor = new JTextField(calcularEspesor(0) + "");

txtEspesor.setColumns(10);

txtEspesor.setBounds(145, 112, 180, 20);

getContentPane().add(txtEspesor);

txtCanOptima = new JTextField(calcularCantidadOptima(0) + "");

txtCanOptima.setColumns(10);

txtCanOptima.setBounds(145, 137, 180, 20);

getContentPane().add(txtCanOptima);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

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

getContentPane().add(btnCerrar);

btnGrabar = new JButton("Grabar");

btnGrabar.addActionListener(this);

btnGrabar.setBounds(335, 36, 89, 23);

getContentPane().add(btnGrabar);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnCerrar(e);

}

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

actionPerformedBtnGrabar(e);

}

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

actionPerformedCboModelo(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

/\*\*

\* Interactuar con el ComboBox

\*/

protected void actionPerformedCboModelo(ActionEvent e) {

double pre, anc, lar, esp;

int mar, can;

mar = cboModelo.getSelectedIndex();

pre = calcularPrecio(mar);

anc = calcularAncho(mar);

lar = calcularLargo(mar);

esp = calcularEspesor(mar);

can = calcularCantidadOptima(mar);

mostrarDatos(pre, anc, lar, esp, can);

}

// Funcion para obtener el indice del comboBox

int calcularModelo() {

return cboModelo.getSelectedIndex();

}

// Funcion para obtener el nombre segun el modelo

String calcularNombre(int mar) {

switch (mar) {

case 0:

return Tienda.modelo0;

case 1:

return Tienda.modelo1;

case 2:

return Tienda.modelo2;

case 3:

return Tienda.modelo3;

default:

return Tienda.modelo4;

}

}

// Funcion para calcular el precio segun el modelo

double calcularPrecio(int mar) {

switch (mar) {

case 0:

return Tienda.precio0;

case 1:

return Tienda.precio1;

case 2:

return Tienda.precio2;

case 3:

return Tienda.precio3;

default:

return Tienda.precio4;

}

}

// Funcion para calcular el ancho segun el modelo

double calcularAncho(int mar) {

switch (mar) {

case 0:

return Tienda.ancho0;

case 1:

return Tienda.ancho1;

case 2:

return Tienda.ancho2;

case 3:

return Tienda.ancho3;

default:

return Tienda.ancho4;

}

}

// Funcion para calcular el largo segun el modelo

double calcularLargo(int mar) {

switch (mar) {

case 0:

return Tienda.largo0;

case 1:

return Tienda.largo1;

case 2:

return Tienda.largo2;

case 3:

return Tienda.largo3;

default:

return Tienda.largo4;

}

}

// Funcion para calcular el espesor segun el modelo

double calcularEspesor(int mar) {

switch (mar) {

case 0:

return Tienda.espesor0;

case 1:

return Tienda.espesor1;

case 2:

return Tienda.espesor2;

case 3:

return Tienda.espesor3;

default:

return Tienda.espesor4;

}

}

// Funcion para calcular la cantidad segun el modelo

int calcularCantidadOptima(int mar) {

switch (mar) {

case 0:

return Tienda.cantidadOptima0;

case 1:

return Tienda.cantidadOptima1;

case 2:

return Tienda.cantidadOptima2;

case 3:

return Tienda.cantidadOptima3;

default:

return Tienda.cantidadOptima4;

}

}

// Funcion para mostrar los datos

void mostrarDatos(double pre, double anc, double lar, double esp, int can) {

txtPrecio.setText(pre + "");

txtAncho.setText(anc + "");

txtLargo.setText(lar + "");

txtEspesor.setText(esp + "");

txtCanOptima.setText(can + "");

}

/\*\*

\* Interactuar con el boton grabar

\*/

protected void actionPerformedBtnGrabar(ActionEvent e) {

// Si el casillero esta vacio saldra un error

if (txtPrecio.getText().length() == 0

|| txtAncho.getText().length() == 0

|| txtLargo.getText().length() == 0

|| txtEspesor.getText().length() == 0

|| txtCanOptima.getText().length() == 0)

JOptionPane.showMessageDialog(this, "Ningun casillero debe estar vacio", "Casillero vacio", 0);

else {

double pre, anc, lar, esp;

int mar, can;

String cad = "";

mar = calcularModelo();

pre = Double.parseDouble(txtPrecio.getText());

anc = Double.parseDouble(txtAncho.getText());

lar = Double.parseDouble(txtLargo.getText());

esp = Double.parseDouble(txtEspesor.getText());

can = Integer.parseInt(txtCanOptima.getText());

if (pre < 0 || anc < 0 || lar < 0 || esp < 0 || can < 0) {

JOptionPane.showMessageDialog(this, "Los valores deben ser mayor a 0", "Valores negativos", 2);

} else {

if (calcularPrecio(mar) != pre) {

cad += mensajePrecio(calcularPrecio(mar), pre);

switch (mar) {

case 0:

Tienda.precio0 = pre;

break;

case 1:

Tienda.precio1 = pre;

break;

case 2:

Tienda.precio2 = pre;

break;

case 3:

Tienda.precio3 = pre;

break;

default:

Tienda.precio4 = pre;

break;

}

}

if (calcularAncho(mar) != anc) {

cad += mensajeAncho(calcularAncho(mar), anc);

switch (mar) {

case 0:

Tienda.ancho0 = anc;

break;

case 1:

Tienda.ancho1 = anc;

break;

case 2:

Tienda.ancho2 = anc;

break;

case 3:

Tienda.ancho3 = anc;

break;

default:

Tienda.ancho4 = anc;

break;

}

}

if (calcularLargo(mar) != lar) {

cad += mensajeLargo(calcularLargo(mar), lar);

switch (mar) {

case 0:

Tienda.largo0 = lar;

break;

case 1:

Tienda.largo1 = lar;

break;

case 2:

Tienda.largo2 = lar;

break;

case 3:

Tienda.largo3 = lar;

break;

default:

Tienda.largo4 = lar;

break;

}

}

if (calcularEspesor(mar) != esp) {

cad += mensajeEspesor(calcularEspesor(mar), esp);

switch (mar) {

case 0:

Tienda.espesor0 = esp;

break;

case 1:

Tienda.espesor1 = esp;

break;

case 2:

Tienda.espesor2 = esp;

break;

case 3:

Tienda.espesor3 = esp;

break;

default:

Tienda.espesor4 = esp;

break;

}

}

if (calcularCantidadOptima(mar) != can) {

cad += mensajeCantidad(calcularCantidadOptima(mar), can);

switch (mar) {

case 0:

Tienda.cantidadOptima0 = can;

break;

case 1:

Tienda.cantidadOptima1 = can;

break;

case 2:

Tienda.cantidadOptima2 = can;

break;

case 3:

Tienda.cantidadOptima3 = can;

break;

default:

Tienda.cantidadOptima4 = can;

break;

}

}

if (cad != "")

mostrarResultados(mensajeModelo(mar) + cad);

}

}

}

String mensajeModelo(int mar) {

return "Modelo: \n" + calcularNombre(mar) + "\n" + "\n";

}

String mensajePrecio(double pa, double pn) {

return "Precio: \n" + pa + " a " + pn + "\n" + "\n";

}

String mensajeAncho(double aa, double an) {

return "Ancho: \n" + aa + " a " + an + "\n" + "\n";

}

String mensajeLargo(double la, double ln) {

return "Largo: \n" + la + " a " + ln + "\n" + "\n";

}

String mensajeEspesor(double ea, double en) {

return "Espesor: \n" + ea + " a " + en + "\n" + "\n";

}

String mensajeCantidad(double ca, double cn) {

return "Can.Óptima: \n" + ca + " a " + cn + "\n" + "\n";

}

void mostrarResultados(String cad) {

JOptionPane.showMessageDialog(this, cad, "Modificacion exitosa!", 1);

}

}

**Listar Ladrillo:**

package cibertec;

import java.awt.EventQueue;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JDialog;

import javax.swing.JScrollPane;

import javax.swing.JTextArea;

import javax.swing.JButton;

public class DialogoListar extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JScrollPane scrollPane;

private JTextArea txtS;

private JButton btnCerrar;

private JButton btnListar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoListar dialog = new DialogoListar();

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

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoListar() {

setModal(true);

setTitle("Listado de Ladrillos");

setBounds(100, 100, 650, 490);

getContentPane().setLayout(null);

scrollPane = new JScrollPane();

scrollPane.setBounds(10, 11, 614, 396);

getContentPane().add(scrollPane);

txtS = new JTextArea();

txtS.setEditable(false);

scrollPane.setViewportView(txtS);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setBounds(225, 417, 89, 23);

getContentPane().add(btnCerrar);

btnListar = new JButton("Listar");

btnListar.addActionListener(this);

btnListar.setBounds(324, 417, 89, 23);

getContentPane().add(btnListar);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnListar(e);

}

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

protected void actionPerformedBtnListar(ActionEvent e) {

txtS.setText("");

imprimir("LISTADO DE LADRILLOS");

mostrarLadrillo(Tienda.modelo0, Tienda.precio0, Tienda.ancho0, Tienda.largo0, Tienda.espesor0, Tienda.cantidadOptima0);

mostrarLadrillo(Tienda.modelo1, Tienda.precio1, Tienda.ancho1, Tienda.largo1, Tienda.espesor1, Tienda.cantidadOptima1);

mostrarLadrillo(Tienda.modelo2, Tienda.precio2, Tienda.ancho2, Tienda.largo2, Tienda.espesor2, Tienda.cantidadOptima2);

mostrarLadrillo(Tienda.modelo3, Tienda.precio3, Tienda.ancho3, Tienda.largo3, Tienda.espesor3, Tienda.cantidadOptima3);

mostrarLadrillo(Tienda.modelo4, Tienda.precio4, Tienda.ancho4, Tienda.largo4, Tienda.espesor4, Tienda.cantidadOptima4);

}

void mostrarLadrillo(String mod, double pre, double anc, double lar, double esp, int can)

{

imprimir("");

imprimir("Modelo : " + mod);

imprimir("Precio : S/" + pre);

imprimir("Ancho (cm) : " + anc + " cm");

imprimir("Largo (cm) : " + lar + " cm");

imprimir("Espesor (mm) : " + esp + " mm");

imprimir("Cantidad Óptima (millares) : " + can);

}

void imprimir(String cad) {

txtS.append(cad + "\n");

}

}

**Vender:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.JComboBox;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JButton;

import javax.swing.JScrollPane;

import javax.swing.JTextArea;

import javax.swing.UIManager;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class DialogoVender extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblModelo;

private JLabel lblPrecio;

private JTextField txtPrecio;

private JLabel lblCantidad;

private JTextField txtCantidad;

private JComboBox<String> cboModelo;

private JButton btnVender;

private JButton btnCerrar;

private JScrollPane scrollPane;

private JTextArea txtS;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoVender dialog = new DialogoVender();

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

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoVender() {

setModal(true);

setTitle("Vender");

setBounds(100, 100, 590, 350);

getContentPane().setLayout(null);

lblModelo = new JLabel("Modelo");

lblModelo.setBounds(15, 15, 50, 14);

getContentPane().add(lblModelo);

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

lblPrecio.setBounds(15, 40, 50, 14);

getContentPane().add(lblPrecio);

txtPrecio = new JTextField();

txtPrecio.setEditable(false);

txtPrecio.setColumns(10);

txtPrecio.setBounds(75, 37, 200, 20);

getContentPane().add(txtPrecio);

lblCantidad = new JLabel("Cantidad");

lblCantidad.setBounds(15, 65, 50, 14);

getContentPane().add(lblCantidad);

txtCantidad = new JTextField();

txtCantidad.setBounds(75, 62, 200, 20);

getContentPane().add(txtCantidad);

txtCantidad.setColumns(10);

cboModelo = new JComboBox<String>();

cboModelo.addActionListener(this);

cboModelo.setModel(new DefaultComboBoxModel<String>(

new String[] { "Ladrillo King Kong 18 Huecos", "Ladrillo King Kong Macizo",

"Ladrillo Pandereta Acanalada", "Ladrillo Patelero", "Ladrillo Techo Hueco" }));

cboModelo.setBounds(75, 11, 200, 22);

getContentPane().add(cboModelo);

btnVender = new JButton("Vender");

btnVender.addActionListener(this);

btnVender.setBounds(471, 11, 89, 23);

getContentPane().add(btnVender);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setBounds(471, 36, 89, 23);

getContentPane().add(btnCerrar);

scrollPane = new JScrollPane();

scrollPane.setBounds(10, 90, 550, 210);

getContentPane().add(scrollPane);

txtS = new JTextArea();

txtS.setEditable(false);

scrollPane.setViewportView(txtS);

txtPrecio.setText(obtenerPrecio(0) + "");

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedCboModelo(e);

}

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

actionPerformedBtnVender(e);

}

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

protected void actionPerformedCboModelo(ActionEvent e) {

int mod = cboModelo.getSelectedIndex();

txtPrecio.setText(obtenerPrecio(mod) + "");

}

double obtenerPrecio(int mod) {

switch (mod) {

case 0:

return Tienda.precio0;

case 1:

return Tienda.precio1;

case 2:

return Tienda.precio2;

case 3:

return Tienda.precio3;

default:

return Tienda.precio4;

}

}

protected void actionPerformedBtnVender(ActionEvent e) {

if (txtCantidad.getText().length() == 0) {

JOptionPane.showMessageDialog(this, "Debes escribir una cantidad antes de realizar una venta",

"Cantidad Vacía", 0);

txtCantidad.requestFocus();

} else if (Integer.parseInt(txtCantidad.getText()) <= 0) {

JOptionPane.showMessageDialog(this, "La cantidad debe ser mayor a 0", "Error Cantidad", 1);

txtCantidad.requestFocus();

txtCantidad.selectAll();

} else {

int mod = cboModelo.getSelectedIndex();

int can = Integer.parseInt(txtCantidad.getText());

double icom = calcularImporteCompra(can);

double ides = calcularImporteDescuento(icom, can);

double ipag = icom - ides;

int uni = calcularCantidadObsequio(can);

mostrarResultados(can, icom, ides, ipag, uni);

aumentarTotales(mod, can, ipag);

Tienda.ventasTotales++;

Tienda.importeTotalAcumulado += ipag;

JOptionPane.showMessageDialog(this, "Venta Nro. " + Tienda.ventasTotales + "\n"

+ "Importe Total Recaudado: S/." + Tienda.importeTotalAcumulado, "Avance de Ventas", 1);

}

}

double calcularImporteCompra(int can) {

return Double.parseDouble(txtPrecio.getText()) \* can;

}

double calcularImporteDescuento(double icom, int can) {

if (can <= 5)

return Tienda.porcentaje1 / 100 \* icom;

else if (can <= 10)

return Tienda.porcentaje2 / 100 \* icom;

else if (can <= 15)

return Tienda.porcentaje3 / 100 \* icom;

else if (can <= 20)

return Tienda.porcentaje4 / 100 \* icom;

else

return Tienda.porcentaje5 / 100 \* icom;

}

int calcularCantidadObsequio(int can) {

if (can <= 5)

return Tienda.obsequioCantidad1;

else if (can <= 10)

return Tienda.obsequioCantidad2;

else if (can <= 15)

return Tienda.obsequioCantidad3;

else if (can <= 20)

return Tienda.obsequioCantidad4;

else

return Tienda.obsequioCantidad5;

}

void aumentarTotales(int mod, int can, double ipag) {

switch (mod) {

case 0:

Tienda.totalVentas0++;

Tienda.totalMillares0 += can;

Tienda.totalVendidos0 += ipag;

break;

case 1:

Tienda.totalVentas1++;

Tienda.totalMillares1 += can;

Tienda.totalVendidos1 += ipag;

break;

case 2:

Tienda.totalVentas2++;

Tienda.totalMillares2 += can;

Tienda.totalVendidos2 += ipag;

break;

case 3:

Tienda.totalVentas3++;

Tienda.totalMillares3 += can;

Tienda.totalVendidos3 += ipag;

break;

default:

Tienda.totalVentas4++;

Tienda.totalMillares4 += can;

Tienda.totalVendidos4 += ipag;

break;

}

}

void mostrarResultados(int can, double icom, double ides, double ipag, int uni) {

txtS.setText("");

txtS.append("BOLETA DE VENTA \n");

txtS.append("\n");

txtS.append("Modelo : " + cboModelo.getSelectedItem() + "\n");

txtS.append("Precio por millar : S/. " + txtPrecio.getText() + "\n");

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

txtS.append("Importe compra : S/. " + icom + "\n");

txtS.append("Importe descuento : S/. " + ides + "\n");

txtS.append("Importe pagar : S/. " + ipag + "\n");

txtS.append("Tipo de obsequio : " + Tienda.tipoObsequio + "\n");

txtS.append("Unidades a obsequiar : " + uni + "\n");

}

}

**Generar reportes:**

package cibertec;

import java.awt.EventQueue;

import java.awt.event.ActionEvent;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import javax.swing.DefaultComboBoxModel;

import javax.swing.UIManager;

import javax.swing.JButton;

import javax.swing.JScrollPane;

import javax.swing.JTextArea;

import java.awt.event.ActionListener;

public class DialogoReportes extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblTipo;

private JComboBox<String> comboBox;

private JButton btnCerrar;

private JScrollPane scrollPane;

private JTextArea txtS;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoReportes dialog = new DialogoReportes();

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

dialog.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoReportes() {

setModal(true);

setTitle("Generar Reportes");

setBounds(100, 100, 650, 420);

getContentPane().setLayout(null);

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

lblTipo.setBounds(15, 15, 80, 14);

getContentPane().add(lblTipo);

comboBox = new JComboBox<String>();

comboBox.setModel(new DefaultComboBoxModel<String>(new String[] { "Ventas por modelo", "Comparación con el precio promedio", "Comparación con la cantidad óptima", "Estadística general" }));

comboBox.setBounds(105, 11, 307, 22);

comboBox.addActionListener(this);

getContentPane().add(comboBox);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setBounds(535, 11, 89, 23);

getContentPane().add(btnCerrar);

scrollPane = new JScrollPane();

scrollPane.setBounds(15, 40, 609, 330);

getContentPane().add(scrollPane);

txtS = new JTextArea();

txtS.setEditable(false);

scrollPane.setViewportView(txtS);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnCerrar(e);

}

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

actionPerformedcomboBox(e);

}

}

// logica boton cerrar

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

//logica cbobox

protected void actionPerformedcomboBox(ActionEvent e) {

int index = obtenerIndex();

tipoIndex(index);

}

//obtener index

public int obtenerIndex() {

return comboBox.getSelectedIndex();

}

//impresion segun el index

public void tipoIndex(int i) {

switch(i) {

case 0:

imprimirVentasModelo();

break;

case 1:

imprimirComparacionPromedio();

break;

case 2:

imprimirCantidadOptima();

break;

default:

estadisticaPrecio();

break;

}

}

//imprimir ventas modelo

public void imprimirVentasModelo() {

double porcent0 = 0;

double porcent1 = 0;

double porcent2 = 0;

double porcent3 = 0;

double porcent4 = 0;

if(Tienda.importeTotalAcumulado == 0) {

porcent0 = 0;

porcent1 = 0;

porcent2 = 0;

porcent3 = 0;

porcent4 = 0;

}else {

porcent0 = (Tienda.totalVendidos0 \* 100) / Tienda.importeTotalAcumulado;

porcent1 = (Tienda.totalVendidos1 \* 100) / Tienda.importeTotalAcumulado;

porcent2 = (Tienda.totalVendidos2 \* 100) / Tienda.importeTotalAcumulado;

porcent3 = (Tienda.totalVendidos3 \* 100) / Tienda.importeTotalAcumulado;

porcent4 = (Tienda.totalVendidos4 \* 100) / Tienda.importeTotalAcumulado;

}

txtS.setText("");

txtS.append("Modelo:.............................. " + Tienda.modelo0 + "\n");

txtS.append("Cantidad de ventas:.................. " + Tienda.totalVentas0 + "\n");

txtS.append("Cantidad de millares vendidos:....... " + Tienda.totalMillares0 + "\n");

txtS.append("Importe total acumulado:............. " + Tienda.totalVendidos0 + "(" + String.format("%.2f", porcent0) + "% del total general)" + "\n" + "\n");

txtS.append("Modelo:.............................. " + Tienda.modelo1 + "\n");

txtS.append("Cantidad de ventas:.................. " + Tienda.totalVentas1 + "\n");

txtS.append("Cantidad de millares vendidos:....... " + Tienda.totalMillares1 + "\n");

txtS.append("Importe total acumulado:............. " + Tienda.totalVendidos1 + "(" + String.format("%.2f", porcent1) + "% del total general)" + "\n" + "\n");

txtS.append("Modelo:.............................. " + Tienda.modelo2 + "\n");

txtS.append("Cantidad de ventas:.................. " + Tienda.totalVentas2 + "\n");

txtS.append("Cantidad de millares vendidos:....... " + Tienda.totalMillares2 + "\n");

txtS.append("Importe total acumulado:............. " + Tienda.totalVendidos2 + "(" + String.format("%.2f", porcent2) + "% del total general)" + "\n" + "\n");

txtS.append("Modelo:.............................. " + Tienda.modelo3 + "\n");

txtS.append("Cantidad de ventas:.................. " + Tienda.totalVentas3 + "\n");

txtS.append("Cantidad de millares vendidos:....... " + Tienda.totalMillares3 + "\n");

txtS.append("Importe total acumulado:............. " + Tienda.totalVendidos3 + "(" + String.format("%.2f", porcent3) + "% del total general)" + "\n" + "\n");

txtS.append("Modelo:.............................. " + Tienda.modelo4 + "\n");

txtS.append("Cantidad de ventas:.................. " + Tienda.totalVentas4 + "\n");

txtS.append("Cantidad de millares vendidos:....... " + Tienda.totalMillares4 + "\n");

txtS.append("Importe total acumulado:............. " + Tienda.totalVendidos4 + "(" + String.format("%.2f", porcent4) + "% del total general)" + "\n" + "\n");

}

public String promedioCadena(double pre, double pro) {

double resta;

if(pre == pro) {

return " igual que el promedio";

}else if(pre > pro) {

resta = pre - pro;

return resta + " mayor que el promedio";

} else {

resta = pro - pre;

return resta + " menor que el promedio";

}

}

public void imprimirComparacionPromedio() {

double prome = (Tienda.precio0 + Tienda.precio1 + Tienda.precio2 + Tienda.precio3 + Tienda.precio4 ) / 5;

String cadenaPromedio0 = promedioCadena(Tienda.precio0, prome);

String cadenaPromedio1 = promedioCadena(Tienda.precio1, prome);

String cadenaPromedio2 = promedioCadena(Tienda.precio2, prome);

String cadenaPromedio3 = promedioCadena(Tienda.precio3, prome);

String cadenaPromedio4 = promedioCadena(Tienda.precio4, prome);

txtS.setText("");

txtS.append("Modelo:.................... " + Tienda.modelo0 + "\n");

txtS.append("Precio:.................... " + Tienda.precio0 + "\n");

txtS.append("Precio promedio:........... " + prome + "\n");

txtS.append("Comparacion:............... " + cadenaPromedio0 + "\n" + "\n");

txtS.append("Modelo:.................... " + Tienda.modelo1 + "\n");

txtS.append("Precio:.................... " + Tienda.precio1 + "\n");

txtS.append("Precio promedio:........... " + prome + "\n");

txtS.append("Comparacion:............... " + cadenaPromedio1 + "\n" + "\n");

txtS.append("Modelo:.................... " + Tienda.modelo2 + "\n");

txtS.append("Precio:.................... " + Tienda.precio2 + "\n");

txtS.append("Precio promedio:........... " + prome + "\n");

txtS.append("Comparacion:............... " + cadenaPromedio2 + "\n" + "\n");

txtS.append("Modelo:.................... " + Tienda.modelo3 + "\n");

txtS.append("Precio:.................... " + Tienda.precio3 + "\n");

txtS.append("Precio promedio:........... " + prome + "\n");

txtS.append("Comparacion:............... " + cadenaPromedio3 + "\n" + "\n");

txtS.append("Modelo:.................... " + Tienda.modelo4 + "\n");

txtS.append("Precio:.................... " + Tienda.precio4 + "\n");

txtS.append("Precio promedio:........... " + prome + "\n");

txtS.append("Comparacion:............... " + cadenaPromedio4 + "\n" + "\n");

}

public String comparacionCadena(int op, int ven) {

int resta;

if(op == ven) {

return " igual que el promedio";

}else if(op > ven) {

resta = op - ven;

return resta + " menor que el promedio";

} else {

resta = ven - op;

return resta + " mayor que el promedio";

}

}

public void imprimirCantidadOptima() {

String optima0 = comparacionCadena(Tienda.cantidadOptima0, Tienda.totalMillares0);

String optima1 = comparacionCadena(Tienda.cantidadOptima1, Tienda.totalMillares1);

String optima2 = comparacionCadena(Tienda.cantidadOptima2, Tienda.totalMillares2);

String optima3 = comparacionCadena(Tienda.cantidadOptima3, Tienda.totalMillares3);

String optima4 = comparacionCadena(Tienda.cantidadOptima4, Tienda.totalMillares4);

txtS.setText("");

txtS.append("Modelo:............................ " + Tienda.modelo0 + "\n");

txtS.append("Cantidad de millares vendidos:..... " + Tienda.totalMillares0 + "\n");

txtS.append("Cantidad optima:................... " + Tienda.cantidadOptima0 + "\n");

txtS.append("Comparacion:....................... " + optima0 + "\n" + "\n");

txtS.append("Modelo:............................ " + Tienda.modelo1 + "\n");

txtS.append("Cantidad de millares vendidos:..... " + Tienda.totalMillares1 + "\n");

txtS.append("Cantidad optima:................... " + Tienda.cantidadOptima1 + "\n");

txtS.append("Comparacion:....................... " + optima1 + "\n" + "\n");

txtS.append("Modelo:............................ " + Tienda.modelo2 + "\n");

txtS.append("Cantidad de millares vendidos:..... " + Tienda.totalMillares2 + "\n");

txtS.append("Cantidad optima:................... " + Tienda.cantidadOptima2 + "\n");

txtS.append("Comparacion:....................... " + optima2 + "\n" + "\n");

txtS.append("Modelo:............................ " + Tienda.modelo3 + "\n");

txtS.append("Cantidad de millares vendidos:..... " + Tienda.totalMillares3 + "\n");

txtS.append("Cantidad optima:................... " + Tienda.cantidadOptima3 + "\n");

txtS.append("Comparacion:....................... " + optima3 + "\n" + "\n");

txtS.append("Modelo:............................ " + Tienda.modelo4 + "\n");

txtS.append("Cantidad de millares vendidos:..... " + Tienda.totalMillares4 + "\n");

txtS.append("Cantidad optima:................... " + Tienda.cantidadOptima4 + "\n");

txtS.append("Comparacion:....................... " + optima4 + "\n" + "\n");

}

public String noVendidoCadena(int ven1, int ven2, int ven3, int ven4, int ven5) {

int noVendido = 0;

if(ven1 == 0) {

noVendido++;

}

if(ven2 == 0) {

noVendido++;

}

if(ven3 == 0) {

noVendido++;

}

if(ven4 == 0) {

noVendido++;

}

if(ven5 == 0) {

noVendido++;

}

return noVendido + "";

}

public void estadisticaPrecio() {

double prome = (Tienda.precio0 + Tienda.precio1 + Tienda.precio2 + Tienda.precio3 + Tienda.precio4 ) / 5;

String cant = noVendidoCadena(Tienda.totalVentas0, Tienda.totalVentas1, Tienda.totalVentas2, Tienda.totalVentas3, Tienda.totalVentas4 );

txtS.setText("");

txtS.append("Precio promedio:...................... " + prome + "\n");

txtS.append("Precio mayor:......................... " + numeroMayor() + "\n");

txtS.append("Precio menor:......................... " + numeroMenor() + "\n");

txtS.append("Cantidad de modelos no vendidos:...... " + cant + "\n" + "\n");

}

public double numeroMayor() {

double numero1 = Tienda.precio0;

double numero2 = Tienda.precio1;

double numero3 = Tienda.precio2;

double numero4 = Tienda.precio3;

double numero5 = Tienda.precio4;

if(numero1 > numero2) {

numero2 = numero1;

}

if(numero2 > numero3) {

numero3 = numero2;

}

if(numero3 > numero4) {

numero4 = numero3;

}

if(numero4 > numero5) {

numero5 = numero4;

}

return numero5;

}

public double numeroMenor() {

double numero1 = Tienda.precio0;

double numero2 = Tienda.precio1;

double numero3 = Tienda.precio2;

double numero4 = Tienda.precio3;

double numero5 = Tienda.precio4;

if(numero1 < numero2) {

numero2 = numero1;

}

if(numero2 < numero3) {

numero3 = numero2;

}

if(numero3 < numero4) {

numero4 = numero3;

}

if(numero4 < numero5) {

numero5 = numero4;

}

return numero5;

}

}

**Configurar descuentos:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.JButton;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.UIManager;

public class configDescuento extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblNewLabel;

private JLabel lblNewLabel\_1;

private JLabel lblNewLabel\_2;

private JLabel lblNewLabel\_3;

private JLabel lblNewLabel\_4;

private JTextField textDesc1;

private JTextField textDesc2;

private JTextField textDesc3;

private JTextField textDesc4;

private JTextField textDesc5;

private JLabel lblNewLabel\_5;

private JLabel lblNewLabel\_6;

private JLabel lblNewLabel\_7;

private JLabel lblNewLabel\_8;

private JLabel lblNewLabel\_9;

private JButton btnAceptar;

private JButton btnCancelar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

configDescuento dialog = new configDescuento();

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

dialog.setVisible(true);

}

catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public configDescuento() {

setModal(true);

setTitle("Configuracion de porcentajes de descuento");

setBounds(100, 100, 434, 171);

getContentPane().setLayout(null);

lblNewLabel = new JLabel("1 a 5 millares");

lblNewLabel.setBounds(10, 11, 98, 14);

getContentPane().add(lblNewLabel);

lblNewLabel\_1 = new JLabel("6 a 10 millares");

lblNewLabel\_1.setBounds(10, 36, 116, 14);

getContentPane().add(lblNewLabel\_1);

lblNewLabel\_2 = new JLabel("11 a 15 millares");

lblNewLabel\_2.setBounds(10, 61, 116, 14);

getContentPane().add(lblNewLabel\_2);

lblNewLabel\_3 = new JLabel("16 a 20 millares");

lblNewLabel\_3.setBounds(10, 86, 116, 14);

getContentPane().add(lblNewLabel\_3);

lblNewLabel\_4 = new JLabel("mas de 20 millares");

lblNewLabel\_4.setBounds(10, 111, 130, 14);

getContentPane().add(lblNewLabel\_4);

textDesc1 = new JTextField();

textDesc1.setText(Tienda.porcentaje1 + "" );

textDesc1.setBounds(150, 8, 86, 20);

getContentPane().add(textDesc1);

textDesc1.setColumns(10);

textDesc2 = new JTextField();

textDesc2.setText(Tienda.porcentaje2 + "" );

textDesc2.setBounds(150, 33, 86, 20);

getContentPane().add(textDesc2);

textDesc2.setColumns(10);

textDesc3 = new JTextField();

textDesc3.setText(Tienda.porcentaje3 + "" );

textDesc3.setBounds(150, 58, 86, 20);

getContentPane().add(textDesc3);

textDesc3.setColumns(10);

textDesc4 = new JTextField();

textDesc4.setText(Tienda.porcentaje4 + "" );

textDesc4.setBounds(150, 83, 86, 20);

getContentPane().add(textDesc4);

textDesc4.setColumns(10);

textDesc5 = new JTextField();

textDesc5.setText(Tienda.porcentaje5 + "" );

textDesc5.setBounds(150, 108, 86, 20);

getContentPane().add(textDesc5);

textDesc5.setColumns(10);

lblNewLabel\_5 = new JLabel("%");

lblNewLabel\_5.setBounds(246, 11, 17, 14);

getContentPane().add(lblNewLabel\_5);

lblNewLabel\_6 = new JLabel("%");

lblNewLabel\_6.setBounds(246, 36, 17, 14);

getContentPane().add(lblNewLabel\_6);

lblNewLabel\_7 = new JLabel("%");

lblNewLabel\_7.setBounds(246, 61, 17, 14);

getContentPane().add(lblNewLabel\_7);

lblNewLabel\_8 = new JLabel("%");

lblNewLabel\_8.setBounds(246, 86, 17, 14);

getContentPane().add(lblNewLabel\_8);

lblNewLabel\_9 = new JLabel("%");

lblNewLabel\_9.setBounds(246, 111, 17, 14);

getContentPane().add(lblNewLabel\_9);

btnAceptar = new JButton("aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setBounds(323, 7, 89, 23);

getContentPane().add(btnAceptar);

btnCancelar = new JButton("cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setBounds(323, 32, 89, 23);

getContentPane().add(btnCancelar);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnProcesar(e);

}

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

protected void actionPerformedBtnProcesar(ActionEvent e) {

cambiarValor();

}

protected void cambiarValor() {

double por1, por2, por3, por4, por5;

String val1, val2, val3, val4, val5;

val1 = textDesc1.getText();

val2 = textDesc2.getText();

val3 = textDesc3.getText();

val4 = textDesc4.getText();

val5 = textDesc5.getText();

if(val1.length() == 0 || val2.length() == 0 || val3.length() == 0|| val4.length() == 0|| val5.length() == 0){

JOptionPane.showMessageDialog(this, "Rellene el espacio");

}else {

por1 = Double.parseDouble(val1);

por2 = Double.parseDouble(val2);

por3 = Double.parseDouble(val3);

por4 = Double.parseDouble(val4);

por5 = Double.parseDouble(val5);

if(por1 < 0 || por2 < 0 || por3 < 0 || por4 < 0 || por5 < 0)

{

JOptionPane.showMessageDialog(this, "Rellene con un numero mayor a 0", "Valor no permitido", 0);

}

else if(por1 > 100 || por2 > 100 || por3 > 100 || por4 > 100 || por5 > 100)

{

JOptionPane.showMessageDialog(this, "Ningun valor puede ser mayor a 100%", "Valor no permitido", 0);

}

else{

Tienda.porcentaje1 = por1;

Tienda.porcentaje2 = por2;

Tienda.porcentaje3 = por3;

Tienda.porcentaje4 = por4;

Tienda.porcentaje5 = por5;

JOptionPane.showMessageDialog(this, "Cambios hechos correctamente", "Valores modificados", 1);

this.dispose();

}

}

}

}

**Configurar obsequios:**

package cibertec;

import java.awt.EventQueue;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JDialog;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.UIManager;

public class configObsequio extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblNewLabel;

private JLabel lblNewLabel\_1;

private JLabel lblNewLabel\_2;

private JLabel lblNewLabel\_3;

private JLabel lblNewLabel\_4;

private JLabel lblNewLabel\_5;

private JTextField textObs0;

private JTextField textObs1;

private JTextField textObs2;

private JTextField textObs3;

private JTextField textObs4;

private JTextField textObs5;

private JButton btnAceptar;

private JButton btnCancelar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

try {

UIManager.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");

} catch (Throwable e) {

e.printStackTrace();

}

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

configObsequio dialog = new configObsequio();

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

dialog.setVisible(true);

}

catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public configObsequio() {

setModal(true);

setTitle("Configurar obsequio");

setBounds(100, 100, 438, 200);

getContentPane().setLayout(null);

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

lblNewLabel.setBounds(10, 11, 98, 14);

getContentPane().add(lblNewLabel);

lblNewLabel\_1 = new JLabel("1 a 5 millares");

lblNewLabel\_1.setBounds(10, 36, 116, 14);

getContentPane().add(lblNewLabel\_1);

lblNewLabel\_2 = new JLabel("6 a 10 millares");

lblNewLabel\_2.setBounds(10, 61, 116, 14);

getContentPane().add(lblNewLabel\_2);

lblNewLabel\_3 = new JLabel("11 a 15 millares");

lblNewLabel\_3.setBounds(10, 86, 116, 14);

getContentPane().add(lblNewLabel\_3);

lblNewLabel\_4 = new JLabel("16 a 20 millares");

lblNewLabel\_4.setBounds(10, 111, 116, 14);

getContentPane().add(lblNewLabel\_4);

lblNewLabel\_5 = new JLabel("mas de 20 millares");

lblNewLabel\_5.setBounds(10, 136, 130, 14);

getContentPane().add(lblNewLabel\_5);

textObs0 = new JTextField();

textObs0.setText(Tienda.tipoObsequio);

textObs0.setBounds(150, 8, 86, 20);

getContentPane().add(textObs0);

textObs0.setColumns(10);

textObs1 = new JTextField();

textObs1.setText(Tienda.obsequioCantidad1 + "" );

textObs1.setBounds(150, 33, 86, 20);

getContentPane().add(textObs1);

textObs1.setColumns(10);

textObs2 = new JTextField();

textObs2.setText(Tienda.obsequioCantidad2 + "" );

textObs2.setBounds(150, 58, 86, 20);

getContentPane().add(textObs2);

textObs2.setColumns(10);

textObs3 = new JTextField();

textObs3.setText(Tienda.obsequioCantidad3 + "" );

textObs3.setBounds(150, 83, 86, 20);

getContentPane().add(textObs3);

textObs3.setColumns(10);

textObs4 = new JTextField();

textObs4.setText(Tienda.obsequioCantidad4 + "" );

textObs4.setBounds(150, 108, 86, 20);

getContentPane().add(textObs4);

textObs4.setColumns(10);

textObs5 = new JTextField();

textObs5.setText(Tienda.obsequioCantidad5 + "" );

textObs5.setBounds(150, 133, 86, 20);

getContentPane().add(textObs5);

textObs5.setColumns(10);

btnAceptar = new JButton("aceptar");

btnAceptar.addActionListener(this);

btnAceptar.setBounds(323, 7, 89, 23);

getContentPane().add(btnAceptar);

btnCancelar = new JButton("cancelar");

btnCancelar.addActionListener(this);

btnCancelar.setBounds(323, 32, 89, 23);

getContentPane().add(btnCancelar);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnProcesar(e);

}

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

protected void actionPerformedBtnProcesar(ActionEvent e) {

cambiarValor();

}

protected void cambiarValor() {

String por1, por2, por3, por4, por5;

por1 = textObs1.getText();

por2 = textObs2.getText();

por3 = textObs3.getText();

por4 = textObs4.getText();

por5 = textObs5.getText();

if(por1.length() == 0 || por2.length() == 0 || por3.length() == 0|| por4.length() == 0|| por5.length() == 0){

JOptionPane.showMessageDialog(this, "Rellene el espacio");

}else{

Tienda.obsequioCantidad1 = Integer.parseInt(por1);

Tienda.obsequioCantidad2= Integer.parseInt(por2);

Tienda.obsequioCantidad3 = Integer.parseInt(por3);

Tienda.obsequioCantidad4 = Integer.parseInt(por4);

Tienda.obsequioCantidad5 = Integer.parseInt(por5);

if(Tienda.obsequioCantidad1 < 0 || Tienda.obsequioCantidad2 < 0 ||Tienda.obsequioCantidad3 < 0 ||Tienda.obsequioCantidad4 < 0 ||Tienda.obsequioCantidad5 < 0){

JOptionPane.showMessageDialog(this, "Rellene con un numero mayor a 0");

}else{

JOptionPane.showMessageDialog(this, "Cambios hechos correctamente!");

this.dispose();

}

}

}

}

**Acerca de:**

package cibertec;

import java.awt.EventQueue;

import javax.swing.JDialog;

import java.awt.Font;

import javax.swing.JSeparator;

import javax.swing.JLabel;

import javax.swing.SwingConstants;

import javax.swing.JButton;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class DialogoAyuda extends JDialog implements ActionListener {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JLabel lblTienda;

private JSeparator separator;

private JLabel lblAutores;

private JLabel lblDavid;

private JLabel lblMauro;

private JLabel lblJesus;

private JLabel lblAndre;

private JLabel lblAngel;

private JLabel lblStephano;

private JButton btnCerrar;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DialogoAyuda dialog = new DialogoAyuda();

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

dialog.setVisible(true);

}

catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the dialog.

\*/

public DialogoAyuda() {

setTitle("Acerca de Tienda");

setModal(true);

setBounds(100, 100, 450, 300);

getContentPane().setLayout(null);

lblTienda = new JLabel("Tienda 1.0");

lblTienda.setHorizontalAlignment(SwingConstants.CENTER);

lblTienda.setFont(new Font("Tahoma", Font.BOLD, 16));

lblTienda.setBounds(174, 10, 86, 20);

getContentPane().add(lblTienda);

separator = new JSeparator();

separator.setBounds(10, 41, 414, 8);

getContentPane().add(separator);

lblAutores = new JLabel("Autores");

lblAutores.setHorizontalAlignment(SwingConstants.CENTER);

lblAutores.setFont(new Font("Tahoma", Font.BOLD, 14));

lblAutores.setBounds(189, 55, 55, 17);

getContentPane().add(lblAutores);

lblDavid = new JLabel("Juan David Anaya (i202312958)");

lblDavid.setHorizontalAlignment(SwingConstants.CENTER);

lblDavid.setBounds(140, 80, 154, 14);

getContentPane().add(lblDavid);

lblMauro = new JLabel("Mauro Arana (i202313409)");

lblMauro.setHorizontalAlignment(SwingConstants.CENTER);

lblMauro.setBounds(152, 100, 129, 14);

getContentPane().add(lblMauro);

lblJesus = new JLabel("Jesús Flores (i202313579)");

lblJesus.setHorizontalAlignment(SwingConstants.CENTER);

lblJesus.setBounds(154, 120, 126, 14);

getContentPane().add(lblJesus);

lblAndre = new JLabel("Andre Pereyra (i202330308)");

lblAndre.setHorizontalAlignment(SwingConstants.CENTER);

lblAndre.setBounds(148, 140, 137, 14);

getContentPane().add(lblAndre);

lblAngel = new JLabel("Angel Pintado (i202314336)");

lblAngel.setHorizontalAlignment(SwingConstants.CENTER);

lblAngel.setBounds(150, 160, 133, 14);

getContentPane().add(lblAngel);

lblStephano = new JLabel("Stephano Yparraguirre (i202313411)");

lblStephano.setHorizontalAlignment(SwingConstants.CENTER);

lblStephano.setBounds(129, 180, 176, 14);

getContentPane().add(lblStephano);

btnCerrar = new JButton("Cerrar");

btnCerrar.addActionListener(this);

btnCerrar.setBounds(172, 227, 89, 23);

getContentPane().add(btnCerrar);

}

public void actionPerformed(ActionEvent e) {

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

actionPerformedBtnCerrar(e);

}

}

protected void actionPerformedBtnCerrar(ActionEvent e) {

this.dispose();

}

}

#### Conclusiones

En este proyecto enfrentamos desafíos y cometimos errores, pero logramos solucionarlo trabajando juntos. Aprendimos la importancia de la comunicación abierta, la responsabilidad y la colaboración. Estamos orgullosos de lo que logramos y esperamos aplicar lo aprendido en futuros proyectos para seguir creciendo como futuros profesionales.

#### Recomendaciones

En términos generales les recomendamos un buena organización debido a que sin esta se encontrarán con muchos problemas mientras más crezca el proyecto sobretodo de escalabilidad y de mantenimiento.

Antes de comenzar a programar hay que definir bien los algoritmos que usaremos y el flujo de programa para asi reducir la cantidad de bugs que puede tener el programa.

Lo que más recomendamos es reutilizar el código y componentizarlo ya que muchas actividades se van a repitir y para poder darle mantenimientos como mejorar u optimizar procesos es muy útil ya que solo se modifica un archivo o un método para afectar a todos los lugares donde este se ejecuta.

#### Bibliografía

https://www.delftstack.com/es/howto/java/sort-an-array-in-java-without-using-sort-method/

https://www.javatpoint.com/java-string-format

https://docs.oracle.com/javase/tutorial/uiswing/events/intro.html

https://stackoverflow.com/questions/31625043/how-to-use-parameters-and-arguments-in-java