**INSTITUTO DE EDUCACIÓN SUPERIOR TECONOLÓGICO CIBERTEC**



**PROYECTO DE ALGORITMOS Y ESTRUCTURA DE DATOS**

**PROYECTO CINE**

**PROFESOR** : CARLOS ROLANDO MENDO BRAVO

## SECCIÓN: T2CM

## SEMESTRE: 2017-1

**INTEGRANTES:**

* Giovanni Ricardo Temoche Clavero (Coordinador)
* Luz Juro Vásquez
* Flavio Rodríguez Valera

Cibertec

Miraflores, mayc

**PRESENTACION**

Este proyecto trata de enfocar desde un punto de vista académico nuestros avances mostrados a lo largo de nuestro segundo ciclo de Educación Superior en el Instituto tecnológico Cibertec.

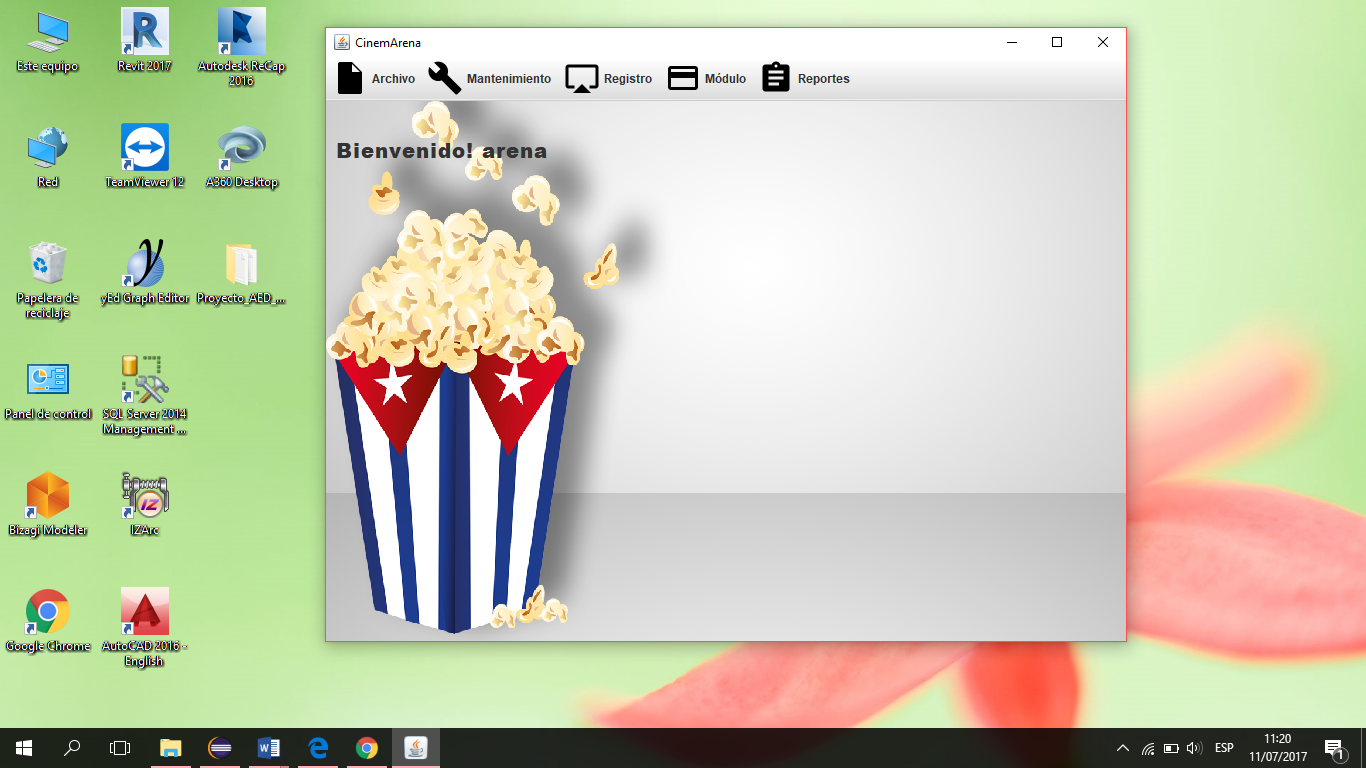
Un Panel es un objeto de la clase JFrame que se utiliza como contenedor de componentes visuales (JButton, JLabel, JTextArea, entre otros) así como de otros paneles. Utilizamos clases ,arreglos lineales e imágenes todo lo aprendido en clase para poder realizar nuestro proyecto Cine.

Al visualizar varias ventanas no tratamos de mostrar varios programas, sino uno solo con varios paneles.

**CUERPO DEL PROYECTO**

* **GUI DE TODO EL PROYECTO (Pantallas Capturadas)**

**1.MENU PRINCIPAL :**



**package** gui;

**import** java.awt.BorderLayout;

**import** java.awt.EventQueue;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.ImageIcon;

**import** javax.swing.JDialog;

**import** javax.swing.JFrame;

**import** javax.swing.JMenu;

**import** javax.swing.JMenuBar;

**import** javax.swing.JMenuItem;

**import** javax.swing.JPanel;

**import** javax.swing.border.EmptyBorder;

**import** arreglo.ArregloUsuario;

**import** javax.swing.JLabel;

**import** java.awt.Font;

**public** **class** Proyecto\_AED\_2017 **extends** JFrame **implements** ActionListener {

**private** JPanelConFondo contentPane;

**private** JMenuBar menuBar = **new** JMenuBar();

**private** JMenu mnArchivo = **new** JMenu("Archivo");

**private** JMenu mnMantenimiento = **new** JMenu("Mantenimiento");

**private** JMenu mnRegistro = **new** JMenu("Registro");

**private** JMenu mnModulo = **new** JMenu("Módulo");

**private** JMenu mnReportes = **new** JMenu("Reportes");

**private** JMenuItem itSalir = **new** JMenuItem("Salir");

**private** JMenuItem itUsuario = **new** JMenuItem("Usuario");

**private** JMenuItem itCliente = **new** JMenuItem("Cliente");

**private** JMenuItem itCancion = **new** JMenuItem("Canción");

**private** JMenuItem itGenero = **new** JMenuItem("Género");

**private** JMenuItem itArtista = **new** JMenuItem("Artista");

**private** JMenuItem itBebida = **new** JMenuItem("Bebida");

**private** JMenuItem itPiqueo = **new** JMenuItem("Piqueo");

**private** JMenuItem itRegPed = **new** JMenuItem("Registro de pedido");

**private** JMenuItem itRegCon = **new** JMenuItem("Registro de consumo");

**private** JMenuItem itPagCon = **new** JMenuItem("Pago de consumo");

**private** JMenuItem itBPedCli = **new** JMenuItem("Buscar pedidos por cliente");

**private** JMenuItem itCanRan = **new** JMenuItem("Canciones más ranqueadas");

**private** JMenuItem itUsuMasPed = **new** JMenuItem("Usuarios con más pedidos");

**private** JMenuItem itComSug = **new** JMenuItem("Comentarios/Sugerencias");

**int** usuarioActual =1001;

**private** JLabel lblBienvenido = **new** JLabel("Bienvenido!");

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

Proyecto\_AED\_2017 frame = **new** Proyecto\_AED\_2017(1001);

frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

**public** Proyecto\_AED\_2017(**int** usuarioActual) {

**this**.usuarioActual=usuarioActual;

setTitle("CinemArena");

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

menuBar.add(mnArchivo);

menuBar.add(mnMantenimiento);

menuBar.add(mnRegistro);

menuBar.add(mnModulo);

menuBar.add(mnReportes);

mnArchivo.add(itSalir);

mnArchivo.setIcon(**new** ImageIcon(Proyecto\_AED\_2017.**class**.getResource("/imagenes/archivo.png")));

mnMantenimiento.add(itUsuario);

mnMantenimiento.add(itCliente);

mnMantenimiento.add(itCancion);

mnMantenimiento.add(itGenero);

mnMantenimiento.add(itArtista);

mnMantenimiento.add(itBebida);

mnMantenimiento.add(itPiqueo);

mnMantenimiento.setIcon(**new** ImageIcon(Proyecto\_AED\_2017.**class**.getResource("/imagenes/mantenimiento.png")));

mnRegistro.add(itRegPed);

mnRegistro.add(itRegCon);

mnRegistro.setIcon(**new** ImageIcon(Proyecto\_AED\_2017.**class**.getResource("/imagenes/registro.png")));

mnModulo.add(itPagCon);

mnModulo.setIcon(**new** ImageIcon(Proyecto\_AED\_2017.**class**.getResource("/imagenes/modulo.png")));

mnReportes.add(itBPedCli);

mnReportes.add(itCanRan);

mnReportes.add(itUsuMasPed);

mnReportes.add(itComSug);

mnReportes.setIcon(**new** ImageIcon(Proyecto\_AED\_2017.**class**.getResource("/imagenes/reportes.png")));

itSalir.addActionListener(**this**);

itUsuario.addActionListener(**this**);

itArtista.addActionListener(**this**);

itBebida.addActionListener(**this**);

itBPedCli.addActionListener(**this**);

itCancion.addActionListener(**this**);

itCanRan.addActionListener(**this**);

itCliente.addActionListener(**this**);

itComSug.addActionListener(**this**);

itGenero.addActionListener(**this**);

itPagCon.addActionListener(**this**);

itPiqueo.addActionListener(**this**);

itRegCon.addActionListener(**this**);

itRegPed.addActionListener(**this**);

itUsuMasPed.addActionListener(**this**);

setJMenuBar(menuBar);

setBounds(100, 100, 1024, 800);

contentPane = **new** JPanelConFondo("/imagenes/cinefondo.png");

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

contentPane.setLayout(**null**);

setContentPane(contentPane);

lblBienvenido.setFont(**new** Font("Arial Black", Font.***BOLD***, 20));

lblBienvenido.setBounds(10, 11, 446, 76);

lblBienvenido.setText("Bienvenido! "+**new** ArregloUsuario().obtenerPorCodigo(usuarioActual).getNombres());

contentPane.add(lblBienvenido);

}

@Override

**public** **void** actionPerformed(ActionEvent arg0) {

// **TODO** Auto-generated method stub

**if** (arg0.getSource() == itSalir){

System.*exit*(0);

}

**else** **if**(arg0.getSource() == itUsuario) {

DUsuario dialog = **new** DUsuario();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itArtista) {

DArtista dialog = **new** DArtista();

}

**else** **if**(arg0.getSource() == itBebida) {

DBebida dialog = **new** DBebida();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itCancion) {

DCancion dialog = **new** DCancion();

}

**else** **if**(arg0.getSource() == itCliente) {

DCliente dialog = **new** DCliente();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itGenero) {

DGenero dialog = **new** DGenero();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itPiqueo) {

DPiqueos dialog = **new** DPiqueos();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itRegPed) {

DRegistroPedido dialog = **new** DRegistroPedido(usuarioActual);

}

**else** **if**(arg0.getSource() == itRegCon) {

DRegistroConsumo dialog = **new** DRegistroConsumo();

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itPagCon) {

DPagoConsumo dialog = **new** DPagoConsumo(1001);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource() == itBPedCli) {

DBuscarPedidoCliente dialog = **new** DBuscarPedidoCliente();

}

**else** **if**(arg0.getSource() == itCanRan) {

DBuscarCancionesRanqueadas dialog = **new** DBuscarCancionesRanqueadas();

}

**else** **if**(arg0.getSource() == itUsuMasPed) {

DBuscarRegistrosMasPedidos dialog = **new** DBuscarRegistrosMasPedidos();

}

**else** {

DSugerencia dialog = **new** DSugerencia(usuarioActual);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

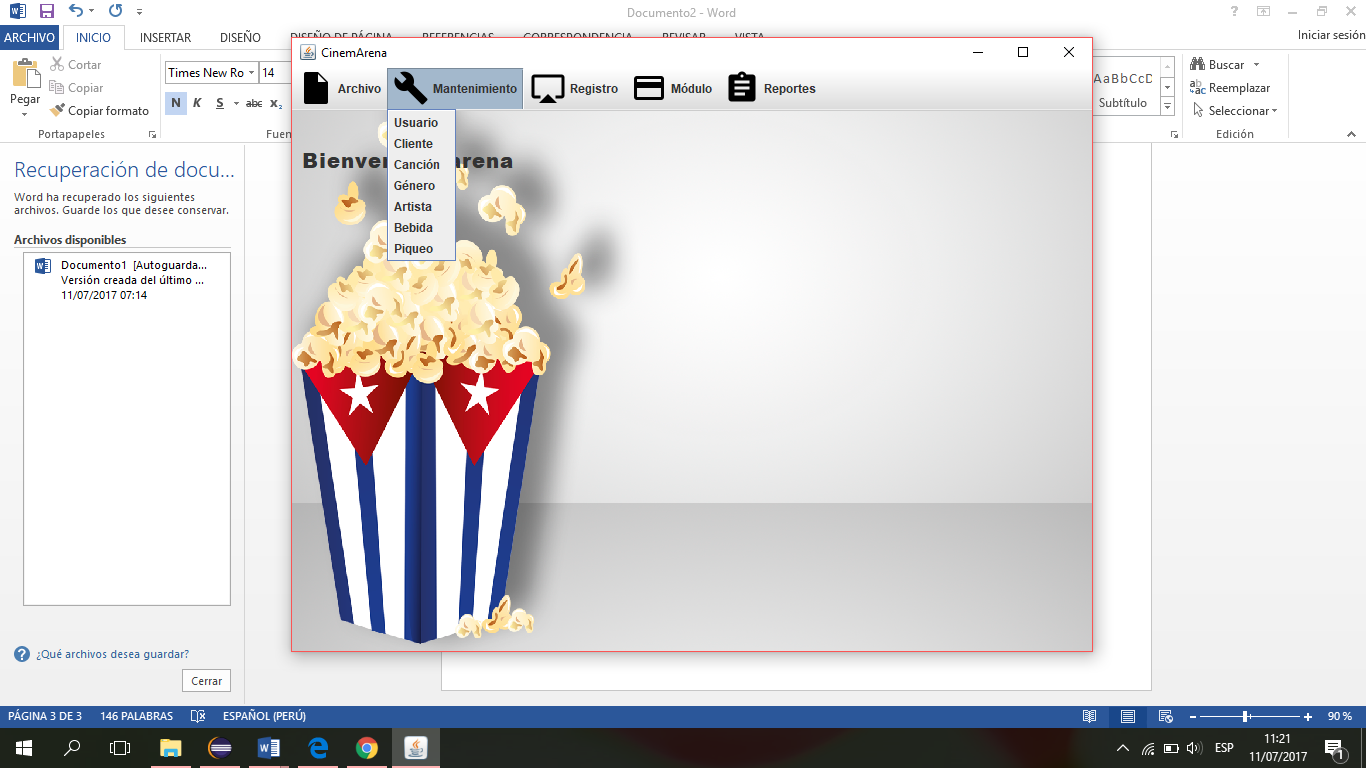
dialog.setVisible(**true**);

}

}

}

**2 .MANTENIMIENTO:**



**USUARIO**

**CLASE USUARIO**

**package** clases;

**public** **class** Usuario {

**public** **static** **final** **int** ***ADM*** = 0;

**public** **static** **final** **int** ***SUP*** = 1;

**public** **static** **final** **int** ***MOZ*** = 2;

**public** **static** **final** **int** ***TAR*** = 0;

**public** **static** **final** **int** ***NOC*** = 1;

**public** **static** **final** String ***PRE*** = "USU";

**public** **static** **final** **int** ***COD\_PRE*** = 1001;

**private** **int** idUsuario;

**private** String apellidoPaterno;

**private** String apellidoMaterno;

**private** String nombres;

**private** **int** tipoEmpleado;

**private** String login;

**private** String password;

**private** **int** turno;

**public** Usuario(**int** idUsuario, String apellidoPaterno, String apellidoMaterno, String nombres,

String login, String password, **int** tipoEmpleado,**int** turno) {

**this**.idUsuario = idUsuario;

**this**.apellidoPaterno = apellidoPaterno;

**this**.apellidoMaterno = apellidoMaterno;

**this**.nombres = nombres;

**this**.login = login;

**this**.password = password;

**this**.tipoEmpleado = tipoEmpleado;

**this**.turno = turno;

}

**public** **int** getIdUsuario() {

**return** idUsuario;

}

**public** **void** setIdUsuario(**int** idUsuario) {

**this**.idUsuario = idUsuario;

}

**public** String getApellidoPaterno() {

**return** apellidoPaterno;

}

**public** **void** setApellidoPaterno(String apellidoPaterno) {

**this**.apellidoPaterno = apellidoPaterno;

}

**public** String getApellidoMaterno() {

**return** apellidoMaterno;

}

**public** **void** setApellidoMaterno(String apellidoMaterno) {

**this**.apellidoMaterno = apellidoMaterno;

}

**public** String getNombres() {

**return** nombres;

}

**public** **void** setNombres(String nombres) {

**this**.nombres = nombres;

}

**public** **int** getTipoEmpleado() {

**return** tipoEmpleado;

}

**public** **void** setTipoEmpleado(**int** tipoEmpleado) {

**this**.tipoEmpleado = tipoEmpleado;

}

**public** String getLogin() {

**return** login;

}

**public** **void** setLogin(String login) {

**this**.login = login;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** **int** getTurno() {

**return** turno;

}

**public** **void** setTurno(**int** turno) {

**this**.turno = turno;

}

//Obtener

**public** **static** String obtenerTipo(**int** tipo){

**switch**(tipo){

**case** ***ADM***:

**return** "Administrador";

**case** ***SUP***:

**return** "Supervisor";

**default**:

**return** "Mozo";

}

}

**public** **static** String obtenerTurno(**int** turno){

**switch**(turno){

**case** ***TAR***:

**return** "Tarde";

**default**:

**return** "Noche";

}

}

}

**ARREGLO USUARIO :**

**package** arreglo;

**import** java.io.BufferedReader;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** java.util.ArrayList;

**import** javax.swing.JOptionPane;

**import** clases.Usuario;

//ATRIBUTO PRIVADO

**public** **class** ArregloUsuario {

**private** String nombreArchivo = "usuarios.txt";

**private** ArrayList<Usuario> usu = **new** ArrayList<Usuario>();

**private** BufferedReader br;

**private** PrintWriter pw;

**public** ArregloUsuario(){

usu = **new** ArrayList<Usuario>();

File archivo = **new** File(nombreArchivo);

**if** (!(archivo.exists())){

insertarPredeterminado();

actualizarArrayList();

}

**else** {

actualizarArrayList();

//actualizarArchivo();

}

}

//OPERACIONES PUBLICAS BASICAS

**public** **int** tamaño(){

**return** usu.size();

}

**public** Usuario obtener(**int** i){

**return** usu.get(i);

}

**public** Usuario obtenerPorCodigo(**int** codigo){

**for**(**int** i=0;i<tamaño();i++){

**if**(obtener(i).getIdUsuario()==codigo){

**return** obtener(i);

}

}

**return** **null**;

}

**public** **void** agregar(Usuario usuario){

usu.add(usuario);

actualizarArchivo();

actualizarArrayList();

}

**public** **void** reemplazar(Usuario usuario, **int** indice){

usu.remove(indice);

usu.add(indice, usuario);

actualizarArchivo();

actualizarArrayList();

}

**public** **void** eliminar(**int** indice){

usu.remove(indice);

actualizarArchivo();

actualizarArrayList();

}

//OBTENER

**public** **int** codigoCorrelativo(){

**if** (usu.size()>0){

**return** usu.get(usu.size()-1).getIdUsuario()+1;

}

**else** {

**return** Usuario.***COD\_PRE***;

}

}

**private** **void** actualizarArrayList() {

usu = **new** ArrayList<Usuario>();

**try**{

br = **new** BufferedReader(**new** FileReader(nombreArchivo));

String linea;

**while** ((linea = br.readLine()) != **null**){

String[] datos = linea.split(";");

**int** codigo = Integer.*parseInt*(datos[0]);

String appat = datos[1];

String apmat = datos[2];

String nombres = datos[3];

String login = datos[4];

String pass = datos[5];

**int** tipo = Integer.*parseInt*(datos[6]);

**int** turno = Integer.*parseInt*(datos[7]);

usu.add(**new** Usuario(codigo, appat, apmat, nombres, login, pass, tipo, turno));

}

br.close();

} **catch** (Exception e) {

**try** {

pw = **new** PrintWriter(**new** FileWriter(nombreArchivo));

} **catch** (FileNotFoundException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

} **catch** (IOException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

}

**private** **void** actualizarArchivo(){

**try** {

pw = **new** PrintWriter(**new** FileWriter(nombreArchivo));

br = **new** BufferedReader(**new** FileReader(nombreArchivo));

**while**(br.readLine()!=**null**){

pw.println("");

}

**for**(**int** i=0;i<usu.size();i++){

pw.println(usu.get(i).getIdUsuario()+";"+

usu.get(i).getApellidoPaterno()+";"+

usu.get(i).getApellidoMaterno()+";"+

usu.get(i).getNombres()+";"+

usu.get(i).getLogin()+";"+

usu.get(i).getPassword()+";"+

usu.get(i).getTipoEmpleado()+";"+

usu.get(i).getTurno());

}

pw.close();

br.close();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

**private** **void** insertarPredeterminado(){

**try** {

pw = **new** PrintWriter(**new** FileWriter(nombreArchivo));

pw.println(Usuario.***COD\_PRE***+";rojas;rojas;arena;arena;bella;0;0");

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

pw.close();

}

}

**GUI DUsuario .java :**

**package** gui;

**import** java.awt.BorderLayout;

**import** java.awt.FlowLayout;

**import** javax.swing.JButton;

**import** javax.swing.JDialog;

**import** javax.swing.JPanel;

**import** javax.swing.border.EmptyBorder;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.table.TableModel;

**import** arreglo.ArregloUsuario;

**import** clases.Usuario;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JTextField;

**import** javax.swing.JComboBox;

**import** javax.swing.DefaultComboBoxModel;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTextArea;

**import** java.awt.event.ActionListener;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** java.awt.event.ActionEvent;

**import** javax.swing.JTable;

**import** javax.swing.JPasswordField;

**public** **class** DUsuario **extends** JDialog **implements** ActionListener, MouseListener {

**private** **static** **final** **int** ***INGRESO*** = 0;

**private** **static** **final** **int** ***CONSULTA*** = 1;

**private** **static** **final** **int** ***MODIFICACION*** = 2;

**private** **final** JPanel contentPanel = **new** JPanel();

**private** JTextField textNombres;

**private** JTextField textApPat;

**private** JTextField textApMat;

**private** JTextField textCodigo;

**private** JComboBox cboTipo;

**private** JComboBox cboTurno;

**private** JButton btnIngreso;

**private** JButton btnConsulta;

**private** JButton btnElimina;

**private** JButton btnListado;

**private** JButton btnOk;

**private** DefaultTableModel modelo = **new** DefaultTableModel();

**private** ArregloUsuario au = **new** ArregloUsuario();

**private** **int** operacionActual;

**private** JTable table;

**private** JLabel lblLogin;

**private** JLabel lblContrasea;

**private** JTextField txtLogin;

**private** JPasswordField txtContraseña;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

**try** {

DUsuario dialog = **new** DUsuario();

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

dialog.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* Create the dialog.

\*/

**public** DUsuario() {

setTitle("Mantenimiento | Usuario");

setBounds(100, 100, 791, 611);

getContentPane().setLayout(**null**);

contentPanel.setBounds(0, 260, 434, 1);

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

getContentPane().add(contentPanel);

contentPanel.setLayout(**null**);

JLabel lblCdigo = **new** JLabel("C\u00F3digo");

lblCdigo.setBounds(21, 29, 108, 14);

getContentPane().add(lblCdigo);

JLabel lblNombres = **new** JLabel("Nombres");

lblNombres.setBounds(21, 54, 108, 14);

getContentPane().add(lblNombres);

textNombres = **new** JTextField();

textNombres.setBounds(154, 51, 125, 20);

getContentPane().add(textNombres);

textNombres.setColumns(10);

JLabel lblApellidoPaterno = **new** JLabel("Apellido paterno");

lblApellidoPaterno.setBounds(21, 79, 108, 14);

getContentPane().add(lblApellidoPaterno);

textApPat = **new** JTextField();

textApPat.setBounds(154, 76, 125, 20);

getContentPane().add(textApPat);

textApPat.setColumns(10);

JLabel lblApellidoMaterno = **new** JLabel("Apellido materno");

lblApellidoMaterno.setBounds(21, 104, 108, 14);

getContentPane().add(lblApellidoMaterno);

textApMat = **new** JTextField();

textApMat.setBounds(154, 101, 125, 20);

getContentPane().add(textApMat);

textApMat.setColumns(10);

JLabel lblTipo = **new** JLabel("Tipo");

lblTipo.setBounds(427, 29, 108, 14);

getContentPane().add(lblTipo);

cboTipo = **new** JComboBox();

cboTipo.setModel(**new** DefaultComboBoxModel(**new** String[] {Usuario.*obtenerTipo*(0),Usuario.*obtenerTipo*(1),

Usuario.*obtenerTipo*(2)}));

cboTipo.setBounds(602, 26, 125, 20);

getContentPane().add(cboTipo);

JLabel lblTurno = **new** JLabel("Turno");

lblTurno.setBounds(427, 54, 108, 14);

getContentPane().add(lblTurno);

cboTurno = **new** JComboBox();

cboTurno.setModel(**new** DefaultComboBoxModel(**new** String[] {Usuario.*obtenerTurno*(0),

Usuario.*obtenerTurno*(1)}));

cboTurno.setBounds(602, 51, 125, 20);

getContentPane().add(cboTurno);

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(10, 136, 743, 379);

getContentPane().add(scrollPane);

modelo.addColumn("CÓDIGO");

modelo.addColumn("APELLIDO PATERNO");

modelo.addColumn("APELLIDO MATERNO");

modelo.addColumn("NOMBRES");

modelo.addColumn("TIPO EMPLEADO");

modelo.addColumn("TURNO");

table = **new** JTable();

table.setFillsViewportHeight(**true**);

table.setModel(modelo);

table.addMouseListener(**this**);

scrollPane.setViewportView(table);

btnIngreso = **new** JButton("Ingreso");

btnIngreso.setBounds(21, 526, 102, 24);

btnIngreso.addActionListener(**this**);

getContentPane().add(btnIngreso);

btnConsulta = **new** JButton("Consulta");

btnConsulta.setBounds(245, 526, 108, 23);

btnConsulta.addActionListener(**this**);

getContentPane().add(btnConsulta);

btnElimina = **new** JButton("Elimina");

btnElimina.setBounds(363, 526, 113, 23);

btnElimina.addActionListener(**this**);

getContentPane().add(btnElimina);

btnListado = **new** JButton("Listado");

btnListado.addActionListener(**this**);

btnListado.setBounds(659, 526, 94, 23);

btnListado.addActionListener(**this**);

getContentPane().add(btnListado);

textCodigo =**new** JTextField();

textCodigo.setBounds(153, 26, 126, 20);

getContentPane().add(textCodigo);

textCodigo.setColumns(10);

//

textCodigo.setEditable(**false**);

btnOk = **new** JButton("OK");

btnOk.setBounds(133, 527, 102, 23);

btnOk.addActionListener(**this**);

btnOk.setEnabled(**false**);

getContentPane().add(btnOk);

lblLogin = **new** JLabel("Login");

lblLogin.setBounds(427, 79, 108, 14);

getContentPane().add(lblLogin);

lblContrasea = **new** JLabel("Contrase\u00F1a");

lblContrasea.setBounds(427, 104, 108, 14);

getContentPane().add(lblContrasea);

txtLogin = **new** JTextField();

txtLogin.setBounds(602, 76, 125, 20);

getContentPane().add(txtLogin);

txtLogin.setColumns(10);

txtContraseña = **new** JPasswordField();

txtContraseña.setBounds(602, 101, 125, 20);

getContentPane().add(txtContraseña);

setResizable(**false**);

modelo.setRowCount(0);

llenarTabla();

}

**private** **void** llenarTabla() {

**for**(**int** i=0;i<au.tamaño();i++){

Object[] datos = {Usuario.***PRE***+au.obtener(i).getIdUsuario(),

au.obtener(i).getApellidoPaterno(),

au.obtener(i).getApellidoMaterno(),

au.obtener(i).getNombres(),

Usuario.*obtenerTipo*(au.obtener(i).getTipoEmpleado()),

Usuario.*obtenerTurno*(au.obtener(i).getTurno())};

modelo.addRow(datos);

}

}

**private** **void** llenarTabla(**int** indice){

Object[] datos = {Usuario.***PRE***+au.obtener(indice).getIdUsuario(),

au.obtener(indice).getApellidoPaterno(),

au.obtener(indice).getApellidoMaterno(),

au.obtener(indice).getNombres(),

Usuario.*obtenerTipo*(au.obtener(indice).getTipoEmpleado()),

Usuario.*obtenerTurno*(au.obtener(indice).getTurno())};

modelo.addRow(datos);

}

**private** **void** llenarTxt(**int** indice) {

**if**(indice>=0){

textCodigo.setText(Usuario.***PRE***+au.obtener(indice).getIdUsuario());

textApPat.setText(au.obtener(indice).getApellidoPaterno());

textApMat.setText(au.obtener(indice).getApellidoMaterno());

textNombres.setText(au.obtener(indice).getNombres());

txtLogin.setText("...");

txtContraseña.setText("...");

cboTipo.setSelectedIndex(au.obtener(indice).getTipoEmpleado());

cboTurno.setSelectedIndex(au.obtener(indice).getTurno());

}

}

**private** **void** vaciarTxt() {

textCodigo.setText("");

textApPat.setText("");

textApMat.setText("");

textNombres.setText("");

txtLogin.setText("");

txtContraseña.setText("");

cboTipo.setSelectedIndex(0);

cboTurno.setSelectedIndex(0);

}

@Override

**public** **void** actionPerformed(ActionEvent arg0) {

**if**(arg0.getSource()==btnIngreso){

textCodigo.setEditable(**false**);

vaciarTxt();

//llenarTabla();

textCodigo.setText(Usuario.***PRE***+au.codigoCorrelativo());

textNombres.requestFocus();

btnOk.setEnabled(**true**);

operacionActual=DUsuario.***INGRESO***;

}

**else** **if**(arg0.getSource()==btnConsulta){

vaciarTxt();

textCodigo.setEditable(**true**);

textCodigo.requestFocus();

btnOk.setEnabled(**true**);

operacionActual=DUsuario.***CONSULTA***;

}

**else** **if**(arg0.getSource()==btnElimina){

//Valido si es mayor o igual que cero para mostrar el mensaje apropiado

**int** indice = table.getSelectedRow();

**if**(indice>=0){

**if**(table.getRowCount()>1){

**int** respuesta = JOptionPane.*showConfirmDialog*(**this**, "¿Desea borrar el registro "+(indice+1)+"?",

"¿Lo tumbo?", JOptionPane.***YES\_NO\_OPTION***, JOptionPane.***QUESTION\_MESSAGE***);

**if** (respuesta==JOptionPane.***YES\_OPTION***){

au.eliminar(indice);

modelo.setRowCount(0);

vaciarTxt();

llenarTabla();

} } **else** {

JOptionPane.*showMessageDialog*(**this**, "Debe haber al menos un usuario", "Qué jesto!", JOptionPane.***ERROR\_MESSAGE***);

}

} **else** {

JOptionPane.*showMessageDialog*(**this**, "No ha seleccionado registro", "Cuidado, mucho cuidado",

JOptionPane.***INFORMATION\_MESSAGE***);

}

}

**else** **if**(arg0.getSource()==btnListado){

vaciarTxt();

modelo.setRowCount(0);

llenarTabla();

}

**else** **if**(arg0.getSource()==btnOk){

**switch**(operacionActual){

**case** DUsuario.***INGRESO***:

okIngreso();

**break**;

**case** DUsuario.***CONSULTA***:

okConsulta();

**break**;

**default**:

okModifcacion();

}

}

}

**private** **void** ingresarRegistro() {

au.agregar(**new** Usuario( au.codigoCorrelativo(),

textApPat.getText(),

textApMat.getText(),

textNombres.getText(),

txtLogin.getText(),

String.*valueOf*(txtContraseña.getPassword()),

cboTipo.getSelectedIndex(),

cboTurno.getSelectedIndex()));

}

**private** **void** reemplazarRegistro(**int** indice) {

**if** (indice>=0){

au.reemplazar(**new** Usuario( au.codigoCorrelativo(),

textApPat.getText(),

textApMat.getText(),

textNombres.getText(),

txtLogin.getText(),

String.*valueOf*(txtContraseña.getPassword()),

cboTipo.getSelectedIndex(),

cboTurno.getSelectedIndex()), indice);

}

}

**private** **int** consultarRegistro() {

//Consulta según todos los campos. Si coincide un valor devuelve el primer registro

**if** (!(textCodigo.getText().equals(""))){

//Si contiene el prefijo intento convertirlo a un número

**if**(textCodigo.getText().contains(Usuario.***PRE***)){

**try**{

**int** codigoUsuario = Integer.*parseInt*(textCodigo.getText().replace(Usuario.***PRE***, ""));

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getIdUsuario()==codigoUsuario){

**return** i;

}

}

} **catch** (Exception e){

e.printStackTrace();

}

}

}

**if** (!(textNombres.getText().equals(""))){

String valor = textNombres.getText();

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getNombres().contains(valor)){

**return** i;

}

}

}

**if** (!(textApPat.getText().equals(""))){

String valor = textApPat.getText();

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getApellidoPaterno().contains(valor)){

**return** i;

}

}

}

**if** (!(textApMat.getText().equals(""))){

String valor = textApMat.getText();

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getApellidoMaterno().contains(valor)){

**return** i;

}

}

}

**else**{

**int** tipo=cboTipo.getSelectedIndex();

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getTipoEmpleado()==tipo){

**return** i;

}

}

**int** turno=cboTurno.getSelectedIndex();

**for** (**int** i=0;i<au.tamaño();i++){

**if**(au.obtener(i).getTurno()==turno){

**return** i;

}

}

}

**return** -1;

}

**private** **void** okModifcacion() {

**if**( textNombres.getText().equals("") ||

textApPat.getText().equals("") ||

textApMat.getText().equals("")){

JOptionPane.*showMessageDialog*( **this**,

"Hay espacios en blanco",

"Cuidado, mucho cuidado",

JOptionPane.***ERROR\_MESSAGE***);

}

**else**{

reemplazarRegistro(table.getSelectedRow());

modelo.setRowCount(0);

llenarTabla();

btnOk.setEnabled(**false**);

}

}

**private** **void** okConsulta() {

**int** registro = consultarRegistro();

**if**(registro !=-1){

llenarTxt(registro);

modelo.setRowCount(0);

llenarTabla(registro);

} **else** {

JOptionPane.*showMessageDialog*(**this**, "No se encontraron registros "

+ "con los criterios de búsqueda", "No encontrado", JOptionPane.***INFORMATION\_MESSAGE***);

}

}

**private** **void** okIngreso() {

**if**( textNombres.getText().equals("") ||

textApPat.getText().equals("") ||

textApMat.getText().equals("") ||

txtLogin.getText().equals("") ||

String.*valueOf*(txtContraseña.getPassword()).equals("")){

JOptionPane.*showMessageDialog*( **this**,

"Hay espacios en blanco",

"Cuidado, mucho cuidado",

JOptionPane.***ERROR\_MESSAGE***);

}

**else**{

ingresarRegistro();

modelo.setRowCount(0);

llenarTabla();

btnOk.setEnabled(**false**);

}

}

@Override

**public** **void** mouseClicked(MouseEvent arg0) {

//Se va a modificar solo si aparecen todas las tablas registradas

**if**(au.tamaño()==table.getRowCount()){

textCodigo.setEditable(**false**);

btnOk.setEnabled(**true**);

llenarTxt(table.getSelectedRow());

operacionActual = DUsuario.***MODIFICACION***;

}

}

@Override

**public** **void** mouseEntered(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseExited(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mousePressed(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

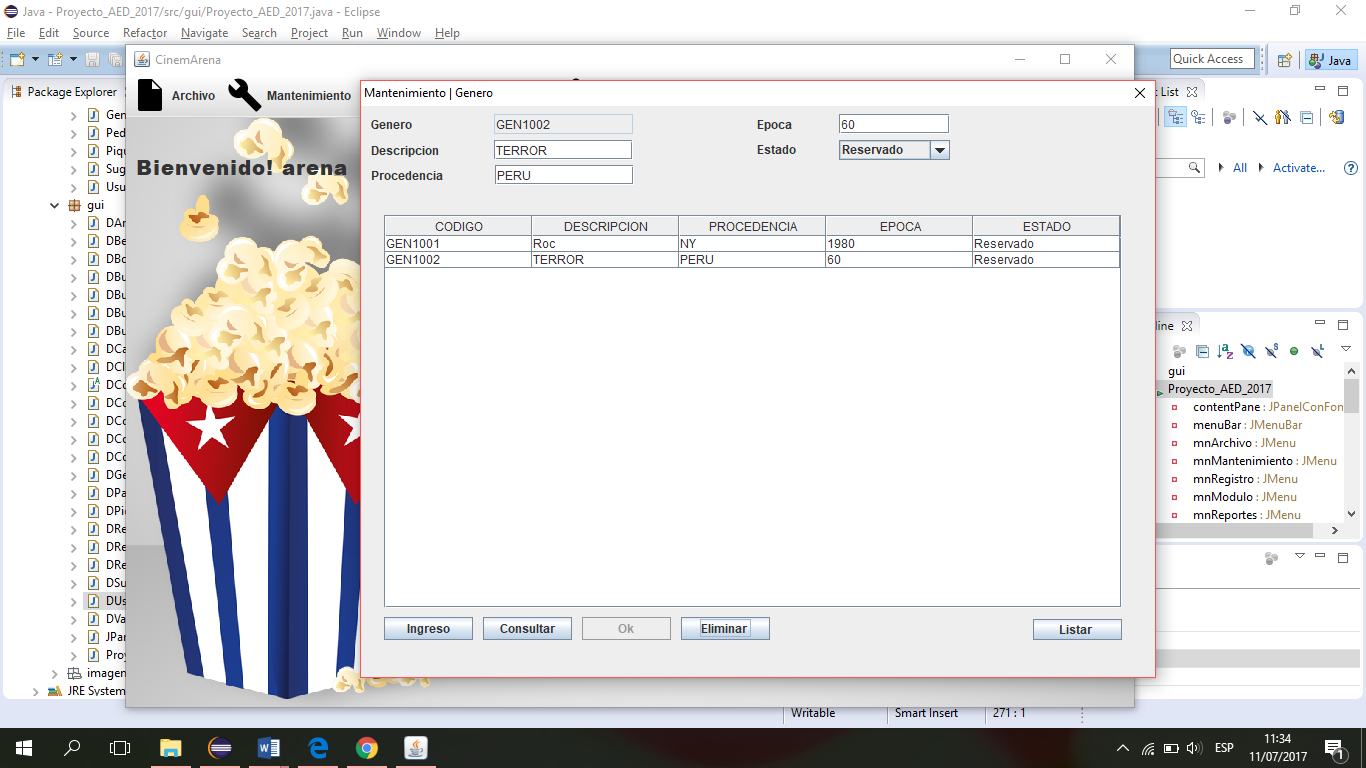
@Override

**public** **void** mouseReleased(MouseEvent arg0) {

// **TODO** Auto-generated method stub

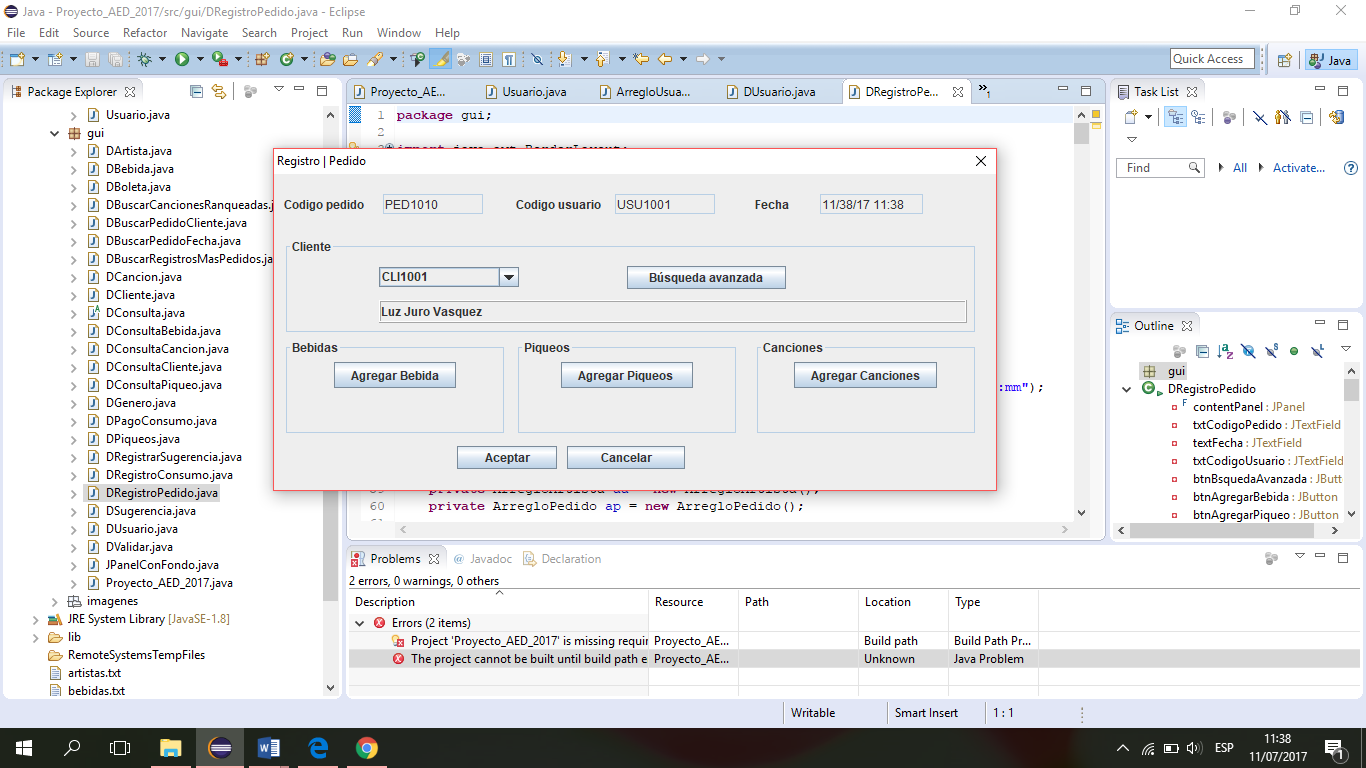
}

}



**REGISTRO PEDIDO:**

* **DRegistroPedido:**



**PEDIDO**

**CLASE**

**package** clases;

**import** java.util.ArrayList;

**public** **class** Pedido {

**public** **static** **final** **int** ***BIC*** = 0;

**public** **static** **final** **int** ***MER*** = 1;

**public** **static** **final** **int** ***AYM*** = 2;

**public** **static** **final** **int** ***AFA*** = 3;

**public** **static** **final** **int** ***DES*** = 4;

**public** **static** **final** **int** ***RES***=0;

**public** **static** **final** **int** ***RAT***=1;

**public** **static** **final** String ***PRE*** = "PED";

**public** **static** **final** **int** ***COD\_PRE*** = 1001;

**private** **int** idPedido;

**private** String fechaPedido;

**private** **int** idUsuario;

**private** **int** idCliente;

**private** ArrayList<Integer> canciones;

**private** ArrayList<Integer> bebidas;

**private** ArrayList<Integer> piqueos;

**private** **int** estado;

**public** Pedido(**int** idPedido, String fechaPedido, **int** idUsuario, **int** idCliente, ArrayList<Integer> canciones, ArrayList<Integer> bebidas, ArrayList<Integer> piqueos,

**int** estado) {

**this**.idPedido = idPedido;

**this**.fechaPedido = fechaPedido;

**this**.idUsuario = idUsuario;

**this**.idCliente = idCliente;

**this**.canciones= canciones;

**this**.bebidas=bebidas;

**this**.piqueos=piqueos;

**this**.estado = estado;

}

// SET/ GET

**public** **int** getIdPedido() {

**return** idPedido;

}

**public** **void** setIdPedido(**int** idPedido) {

**this**.idPedido = idPedido;

}

**public** String getFechaPedido() {

**return** fechaPedido;

}

**public** **void** setFechaPedido(String fechaPedido) {

**this**.fechaPedido = fechaPedido;

}

**public** **int** getIdUsuario() {

**return** idUsuario;

}

**public** **void** setIdUsuario(**int** idUsuario) {

**this**.idUsuario = idUsuario;

}

**public** **int** getIdCliente() {

**return** idCliente;

}

**public** **void** setIdCliente(**int** idCliente) {

**this**.idCliente = idCliente;

}

**public** ArrayList<Integer> getCanciones() {

**return** canciones;

}

**public** **void** setCanciones(ArrayList<Integer> canciones) {

**this**.canciones = canciones;

}

**public** ArrayList<Integer> getBebidas() {

**return** bebidas;

}

**public** **void** setBebidas(ArrayList<Integer> bebidas) {

**this**.bebidas = bebidas;

}

**public** ArrayList<Integer> getPiqueos() {

**return** piqueos;

}

**public** **void** setPiqueos(ArrayList<Integer> piqueos) {

**this**.piqueos = piqueos;

}

**public** **int** getEstado() {

**return** estado;

}

**public** **void** setEstado(**int** estado) {

**this**.estado = estado;

}

//OBTENER

**public** **static** String obtenerCanciones(**int** canciones){

**switch**(canciones){

**case** ***BIC***:

**return** "La Bicicleta";

**case** ***MER***:

**return** " Me Rehuso";

**case** ***AYM***:

**return** "Ay mi dios";

**case** ***AFA***:

**return** "Al filo de tu amor";

**default**:

**return** "Despacito ";

}

}

**public** **static** String obtenerEstado(**int** estados){

**switch**(estados){

**case** ***RES***:

**return** "Reservado";

**default**:

**return** "Reserva Atendida";

}

}

}

**ARREGLO**

**package** arreglo;

**import** java.io.BufferedReader;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** java.util.ArrayList;

**import** javax.swing.JOptionPane;

**import** clases.Pedido;

**import** clases.Piqueo;

**import** clases.Pedido;

**import** gui.DPiqueos;

//ATRIBUTO PRIVADO

**public** **class** ArregloPedido {

**private** String nombreArchivo = "pedidos.txt";

**private** ArrayList<Pedido> ped = **new** ArrayList<Pedido>();

**private** BufferedReader br;

**private** PrintWriter pw;

**public** ArregloPedido(){

ped = **new** ArrayList<Pedido>();

File archivo = **new** File(nombreArchivo);

**if** (!(archivo.exists())){

actualizarArrayList();

}

**else** {

actualizarArrayList();

actualizarArchivo();

}

}

//OPERACIONES PUBLICAS BASICAS

**public** **int** tamaño(){

**return** ped.size();

}

**public** Pedido obtener(**int** i){

**return** ped.get(i);

}

**public** **void** agregar(Pedido pedido){

ped.add(pedido);

actualizarArchivo();

actualizarArrayList();

}

**public** **void** reemplazar(Pedido pedido, **int** indice){

ped.remove(indice);

ped.add(indice, pedido);

actualizarArchivo();

actualizarArrayList();

}

**public** **void** eliminar(**int** indice){

ped.remove(indice);

actualizarArchivo();

actualizarArrayList();

}

//OBTENER

**public** **int** codigoCorrelativo(){

**if** (ped.size()>0){

**return** ped.get(ped.size()-1).getIdPedido()+1;

}

**else** {

**return** Pedido.***COD\_PRE***;

}

}

**private** **void** actualizarArrayList() {

ped = **new** ArrayList<Pedido>();

**try**{

br = **new** BufferedReader(**new** FileReader(nombreArchivo));

String linea;

**while** ((linea = br.readLine()) != **null**){

String[] datos = linea.split(";");

**int** codigo = Integer.*parseInt*(datos[0]);

String fecha = datos[1];

**int** idUsuario = Integer.*parseInt*(datos[2]);

**int** idCliente = Integer.*parseInt*(datos[3]);

**int** numCanciones = Integer.*parseInt*(datos[4]);

ArrayList<Integer> canciones = **new** ArrayList<Integer>();

//Obtengo las canciones

**for**(**int** i=0;i<numCanciones;i++){

canciones.add(Integer.*parseInt*(datos[5+i]));

}

**int** numBebidas = Integer.*parseInt*(datos[5+numCanciones]);

ArrayList<Integer> bebidas = **new** ArrayList<Integer>();

//Obtengo las bebidas

**for**(**int** i=0;i<numBebidas;i++){

bebidas.add(Integer.*parseInt*(datos[6+numCanciones+i]));

}

**int** numPiqueos = Integer.*parseInt*(datos[6+numCanciones+numBebidas]);

ArrayList<Integer> piqueos = **new** ArrayList<Integer>();

//Obtengo los piqueos

**for**(**int** i=0;i<numPiqueos;i++){

piqueos.add(Integer.*parseInt*(datos[7+numCanciones+numBebidas+i]));

}

**int** estado = Integer.*parseInt*(datos[7+numCanciones+numBebidas+numPiqueos]);

//System.out.println(codigo +";"+ fecha+";"+ idUsuario+";" +idCliente+";"+ canciones+";"+ bebidas+";"+ piqueos+";"+ estado);

//System.out.println(numCanciones+";"+numBebidas+";"+numPiqueos);

ped.add(**new** Pedido(codigo, fecha, idUsuario, idCliente, canciones, bebidas, piqueos, estado));

}

br.close();

} **catch** (Exception e) {

**try** {

pw = **new** PrintWriter(**new** FileWriter(nombreArchivo));

} **catch** (FileNotFoundException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

} **catch** (IOException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

}

**private** **void** actualizarArchivo(){

**try** {

pw = **new** PrintWriter(**new** FileWriter(nombreArchivo));

br = **new** BufferedReader(**new** FileReader(nombreArchivo));

**while**(br.readLine()!=**null**){

pw.println("");

}

//Los archivos tendrán esta estructura:

//idPedido;fechaPedido;idUsuario;numeroCanciones;cancion[1-n];numeroBebidas...

**for**(**int** i=0;i<ped.size();i++){

//Obtengo las canciones

StringBuilder canciones = **new** StringBuilder();

**for**(**int** a=0;a<ped.get(i).getCanciones().size();a++){

canciones.append((ped.get(i).getCanciones().get(a)));

canciones.append(";");

}

//Obtengo las bebidas

StringBuilder bebidas = **new** StringBuilder();

**for**(**int** a=0;a<ped.get(i).getBebidas().size();a++){

bebidas.append((ped.get(i).getBebidas().get(a)));

bebidas.append(";");

}

//Obtengo los piqueos

StringBuilder piqueos = **new** StringBuilder();

**for**(**int** a=0;a<ped.get(i).getPiqueos().size();a++){

piqueos.append((ped.get(i).getPiqueos().get(a)));

piqueos.append(";");

}

pw.println(ped.get(i).getIdPedido()+";"+

ped.get(i).getFechaPedido()+";"+

ped.get(i).getIdUsuario()+";"+

ped.get(i).getIdCliente()+";"+

ped.get(i).getCanciones().size()+";"+

canciones.toString()+

ped.get(i).getBebidas().size()+";"+

bebidas.toString()+

ped.get(i).getPiqueos().size()+";"+

piqueos.toString()+

ped.get(i).getEstado());

}

pw.close();

br.close();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

**DREGISTROPEDIDO :**

**package** gui;

**import** java.awt.BorderLayout;

**import** java.awt.FlowLayout;

**import** javax.swing.JButton;

**import** javax.swing.JDialog;

**import** javax.swing.JPanel;

**import** javax.swing.border.BevelBorder;

**import** javax.swing.border.EmptyBorder;

**import** javax.swing.border.EtchedBorder;

**import** javax.swing.border.TitledBorder;

**import** arreglo.ArregloArtista;

**import** arreglo.ArregloBebida;

**import** arreglo.ArregloCancion;

**import** arreglo.ArregloCliente;

**import** arreglo.ArregloPedido;

**import** arreglo.ArregloPiqueo;

**import** clases.Cliente;

**import** clases.Pedido;

**import** clases.Usuario;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JTextField;

**import** javax.swing.SpringLayout;

**import** javax.swing.JComboBox;

**import** java.awt.event.ActionListener;

**import** java.text.SimpleDateFormat;

**import** java.util.ArrayList;

**import** java.util.Date;

**import** java.awt.event.ActionEvent;

**import** javax.swing.BorderFactory;

**import** javax.swing.BoxLayout;

**import** javax.swing.JScrollPane;

**import** javax.swing.DefaultComboBoxModel;

**import** javax.swing.ImageIcon;

**public** **class** DRegistroPedido **extends** JDialog **implements** ActionListener{

**private** **final** JPanel contentPanel = **new** JPanel();

**private** JTextField txtCodigoPedido;

**private** JTextField textFecha;

**private** JTextField txtCodigoUsuario;

**private** JButton btnBsquedaAvanzada;

**private** JButton btnAgregarBebida;

**private** JButton btnAgregarPiqueo;

**private** JButton btnAgregarCancion;

**private** JButton btnAceptar;

**private** JButton btnCancelar;

**private** SimpleDateFormat dateFormat = **new** SimpleDateFormat("dd/mm/yy hh:mm");

**private** String fecha = dateFormat.format(**new** Date());

**private** ArregloCliente acl = **new** ArregloCliente();

**private** ArregloBebida ab = **new** ArregloBebida();

**private** ArregloPiqueo api = **new** ArregloPiqueo();

**private** ArregloCancion aca = **new** ArregloCancion();

**private** ArregloArtista aa = **new** ArregloArtista();

**private** ArregloPedido ap = **new** ArregloPedido();

**private** String[] nombresIdClientes;

**private** ArrayList<JButton> botonesBebidas = **new** ArrayList<JButton>();

**private** ArrayList<JPanel> panelesBebidas = **new** ArrayList<JPanel>();

**private** ArrayList<JButton> botonesPiqueos = **new** ArrayList<JButton>();

**private** ArrayList<JPanel> panelesPiqueos = **new** ArrayList<JPanel>();

**private** ArrayList<JButton> botonesCanciones = **new** ArrayList<JButton>();

**private** ArrayList<JPanel> panelesCanciones = **new** ArrayList<JPanel>();

//Modificables

JComboBox<String> cboCliente;

JLabel lblDatosCliente;

JPanel panelBebidas;

JPanel panelPiqueos;

JPanel panelCanciones;

//Seleccionados

**private** **int** usuarioActual;

**int** clienteActual;

ArrayList<Integer> bebidasSeleccionadas = **new** ArrayList<Integer>();

ArrayList<Integer> piqueosSeleccionados = **new** ArrayList<Integer>();

ArrayList<Integer> cancionesSeleccionadas = **new** ArrayList<Integer>();

**int** codigoCorrelativo=ap.codigoCorrelativo();

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

**try** {

DRegistroPedido dialog = **new** DRegistroPedido(1001);

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

dialog.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* Create the dialog.

\*/

**public** DRegistroPedido(**int** usuarioActual) {

//Valido que existan bebidas, piqueos y canciones

**if** (ab.tamaño()!=0 && api.tamaño()!=0 && aca.tamaño()!=0){

**this**.usuarioActual=usuarioActual;

setTitle("Registro | Pedido");

setBounds(100, 100, 728, 345);

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

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

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

contentPanel.setLayout(**null**);

llenarComboIdClientes();

JLabel lblCodigoPedido = **new** JLabel("Codigo pedido");

lblCodigoPedido.setBounds(10, 23, 89, 14);

contentPanel.add(lblCodigoPedido);

JLabel label = **new** JLabel("Fecha");

label.setBounds(481, 23, 55, 14);

contentPanel.add(label);

txtCodigoPedido = **new** JTextField();

txtCodigoPedido.setEditable(**false**);

txtCodigoPedido.setBounds(109, 20, 100, 20);

txtCodigoPedido.setColumns(10);

txtCodigoPedido.setText(Pedido.***PRE***+codigoCorrelativo);

contentPanel.add(txtCodigoPedido);

txtCodigoUsuario = **new** JTextField();

txtCodigoUsuario.setEditable(**false**);

txtCodigoUsuario.setColumns(10);

txtCodigoUsuario.setBounds(341, 20, 100, 20);

txtCodigoUsuario.setText(Usuario.***PRE***+usuarioActual);

contentPanel.add(txtCodigoUsuario);

textFecha = **new** JTextField();

textFecha.setEditable(**false**);

textFecha.setBounds(546, 20, 103, 20);

textFecha.setText(fecha);

contentPanel.add(textFecha);

textFecha.setColumns(10);

btnCancelar = **new** JButton("Cancelar");

btnCancelar.setBounds(293, 272, 118, 23);

btnCancelar.addActionListener(**this**);

contentPanel.add(btnCancelar);

btnAceptar = **new** JButton("Aceptar");

btnAceptar.setBounds(183, 272, 100, 23);

btnAceptar.addActionListener(**this**);

contentPanel.add(btnAceptar);

JPanel panelCliente = **new** JPanel();

panelCliente.setBounds(10, 64, 693, 96);

panelCliente.setBorder(BorderFactory.*createTitledBorder*("Cliente"));

contentPanel.add(panelCliente);

panelCliente.setLayout(**null**);

cboCliente = **new** JComboBox<String>();

cboCliente.setModel(**new** DefaultComboBoxModel<String>(nombresIdClientes));

cboCliente.setBounds(95, 29, 140, 20);

cboCliente.addActionListener(**this**);

panelCliente.add(cboCliente);

btnBsquedaAvanzada = **new** JButton("B\u00FAsqueda avanzada");

btnBsquedaAvanzada.setBounds(343, 28, 159, 23);

btnBsquedaAvanzada.addActionListener(**this**);

panelCliente.add(btnBsquedaAvanzada);

lblDatosCliente = **new** JLabel();

lblDatosCliente.setBounds(95, 62, 588, 23);

lblDatosCliente.setBorder(BorderFactory.*createEtchedBorder*(EtchedBorder.***RAISED***));

panelCliente.add(lblDatosCliente);

JScrollPane scrollPaneBebidas = **new** JScrollPane();

scrollPaneBebidas.setBounds(10, 165, 222, 96);

scrollPaneBebidas.setBorder(BorderFactory.*createTitledBorder*("Bebidas"));

contentPanel.add(scrollPaneBebidas);

panelBebidas = **new** JPanel();

panelBebidas.setLayout(**new** BoxLayout(panelBebidas, BoxLayout.***Y\_AXIS***));

scrollPaneBebidas.setViewportView(panelBebidas);

JPanel panelBtnAgregarBebidas = **new** JPanel();

panelBtnAgregarBebidas.setLayout(**new** FlowLayout());

//panelBtnAgregarBebidas.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));

panelBebidas.add(panelBtnAgregarBebidas);

btnAgregarBebida = **new** JButton("Agregar Bebida");

btnAgregarBebida.addActionListener(**this**);

panelBtnAgregarBebidas.add(btnAgregarBebida);

JScrollPane scrollPanePiqueos = **new** JScrollPane();

scrollPanePiqueos.setBounds(242, 165, 222, 96);

scrollPanePiqueos.setBorder(BorderFactory.*createTitledBorder*("Piqueos"));

contentPanel.add(scrollPanePiqueos);

panelPiqueos = **new** JPanel();

panelPiqueos.setLayout(**new** BoxLayout(panelPiqueos, BoxLayout.***Y\_AXIS***));

scrollPanePiqueos.setViewportView(panelPiqueos);

JPanel panelBtnAgregarPiqueos = **new** JPanel();

panelBtnAgregarPiqueos.setLayout(**new** FlowLayout());

//panelBtnAgregarPiqueos.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));

panelPiqueos.add(panelBtnAgregarPiqueos);

btnAgregarPiqueo = **new** JButton("Agregar Piqueos ");

btnAgregarPiqueo.addActionListener(**this**);

panelBtnAgregarPiqueos.add(btnAgregarPiqueo);

JScrollPane scrollPaneCanciones = **new** JScrollPane();

scrollPaneCanciones.setBorder(BorderFactory.*createTitledBorder*("Canciones"));

scrollPaneCanciones.setBounds(481, 165, 222, 96);

contentPanel.add(scrollPaneCanciones);

panelCanciones = **new** JPanel();

panelCanciones.setLayout(**new** BoxLayout(panelCanciones, BoxLayout.***Y\_AXIS***));

scrollPaneCanciones.setViewportView(panelCanciones);

JPanel panelBtnAgregarCanciones = **new** JPanel();

panelBtnAgregarCanciones.setLayout(**new** FlowLayout());

panelCanciones.add(panelBtnAgregarCanciones);

btnAgregarCancion = **new** JButton("Agregar Canciones");

btnAgregarCancion.addActionListener(**this**);

panelBtnAgregarCanciones.add(btnAgregarCancion);

JLabel lblCodigoUsuario = **new** JLabel("Codigo usuario");

lblCodigoUsuario.setBounds(242, 23, 89, 14);

contentPanel.add(lblCodigoUsuario);

mostrarNombresCliente(cboCliente.getSelectedIndex());

setResizable(**false**);

setLocationRelativeTo(**this**);

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

setModal(**true**);

setVisible(**true**);

} **else** {

**this**.dispose();

JOptionPane.*showMessageDialog*(**this**, "Necesita tener como mínimo una bebida, un piqueo y una canción", "Cuidado, mucho cuidado",

JOptionPane.***ERROR\_MESSAGE***);

}

}

**private** **void** llenarComboIdClientes() {

nombresIdClientes = **new** String[acl.tamaño()];

**for**(**int** i=0;i<acl.tamaño();i++){

nombresIdClientes[i] = Cliente.***PRE***+acl.obtener(i).getIdCliente();

}

}

**void** mostrarNombresCliente(**int** selectedIndex) {

String datos = acl.obtener(selectedIndex).getNombres()+" "+

acl.obtener(selectedIndex).getApellidoPaterno()+" "+

acl.obtener(selectedIndex).getApellidoMaterno();

lblDatosCliente.setText(datos);

}

**void** agregarBebidas(String datosBebida){

//Agrego el panel con el botón eliminar

JPanel panel = **new** JPanel();

panel.setLayout(**new** FlowLayout());

panel.add(**new** JLabel(datosBebida));

panel.setBorder(BorderFactory.*createEtchedBorder*(EtchedBorder.***RAISED***));

panelesBebidas.add(panel);

//Agrego el boton al ArrayList

JButton boton = **new** JButton();

boton.setIcon(**new** ImageIcon(DRegistroPedido.**class**.getResource("/imagenes/eliminar.png")));

boton.addActionListener(**this**);

botonesBebidas.add(boton);

panel.add(boton);

panelBebidas.add(panel);

validate();

}

**void** agregarPiqueos(String datosPiqueo){

//Agrego el panel con el botón eliminar

JPanel panel = **new** JPanel();

panel.setLayout(**new** FlowLayout());

panel.add(**new** JLabel(datosPiqueo));

panel.setBorder(BorderFactory.*createEtchedBorder*(EtchedBorder.***RAISED***));

panelesPiqueos.add(panel);

//Agrego el boton al ArrayList

JButton boton = **new** JButton();

boton.setIcon(**new** ImageIcon(DRegistroPedido.**class**.getResource("/imagenes/eliminar.png")));

boton.addActionListener(**this**);

botonesPiqueos.add(boton);

panel.add(boton);

panelPiqueos.add(panel);

validate();

}

**void** agregarCanciones(String datosCancion){

//Agrego el panel con el botón eliminar

JPanel panel = **new** JPanel();

panel.setLayout(**new** FlowLayout());

panel.add(**new** JLabel(datosCancion));

panel.setBorder(BorderFactory.*createEtchedBorder*(EtchedBorder.***RAISED***));

panelesCanciones.add(panel);

//Agrego el boton al ArrayList

JButton boton = **new** JButton();

boton.setIcon(**new** ImageIcon(DRegistroPedido.**class**.getResource("/imagenes/eliminar.png")));

boton.addActionListener(**this**);

botonesCanciones.add(boton);

panel.add(boton);

panelCanciones.add(panel);

validate();

}

@Override

**public** **void** actionPerformed(ActionEvent arg0) {

**if** (arg0.getSource()==cboCliente){

//Método para mostrar los datos del cliente en un JLabel

clienteActual = cboCliente.getSelectedIndex();

mostrarNombresCliente(clienteActual);

}

**else** **if** (arg0.getSource()==btnBsquedaAvanzada){

DConsultaCliente dialog = **new** DConsultaCliente(acl, **this**);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource()==btnAgregarBebida){

DConsultaBebida dialog = **new** DConsultaBebida(ab, **this**);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource()==btnAgregarPiqueo){

DConsultaPiqueo dialog = **new** DConsultaPiqueo(api, **this**);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

**else** **if**(arg0.getSource()==btnAgregarCancion){

DConsultaCancion dialog = **new** DConsultaCancion(aca, aa, **this**);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

//Control de eliminación

**else** **if**(botonesBebidas.contains(arg0.getSource())){

**int** indice = botonesBebidas.indexOf(arg0.getSource());

**int** respuesta = JOptionPane.*showConfirmDialog*(**this**, "¿Quieres borrar la bebida "+(indice+1)+"?", "¿En serio?", JOptionPane.***YES\_NO\_OPTION***);

**if** (respuesta == JOptionPane.***YES\_OPTION***){

bebidasSeleccionadas.remove(indice);

panelesBebidas.remove(indice);

botonesBebidas.remove(indice);

panelBebidas.remove(indice+1);

validate();

}

}

**else** **if**(botonesPiqueos.contains(arg0.getSource())){

**int** indice = botonesPiqueos.indexOf(arg0.getSource());

**int** respuesta = JOptionPane.*showConfirmDialog*(**this**, "¿Quieres borrar el piqueo "+(indice+1)+"?", "¿En serio?", JOptionPane.***YES\_NO\_OPTION***);

**if** (respuesta == JOptionPane.***YES\_OPTION***){

piqueosSeleccionados.remove(indice);

panelesPiqueos.remove(indice);

botonesPiqueos.remove(indice);

panelPiqueos.remove(indice+1);

validate();

}

}

**else** **if**(botonesCanciones.contains(arg0.getSource())){

**int** indice = botonesCanciones.indexOf(arg0.getSource());

**int** respuesta = JOptionPane.*showConfirmDialog*(**this**, "¿Quieres borrar la canción "+(indice+1)+"?", "¿En serio?", JOptionPane.***YES\_NO\_OPTION***);

**if** (respuesta == JOptionPane.***YES\_OPTION***){

cancionesSeleccionadas.remove(indice);

panelesCanciones.remove(indice);

botonesCanciones.remove(indice);

panelCanciones.remove(indice+1);

validate();

}

}

**else** **if**(arg0.getSource()==btnAceptar){

ap.agregar(**new** Pedido(codigoCorrelativo, fecha, usuarioActual, clienteActual, cancionesSeleccionadas, bebidasSeleccionadas, piqueosSeleccionados, Pedido.***RES***));

**this**.dispose();

}

**else** {

**int** respuesta = JOptionPane.*showConfirmDialog*(**this**, "¿Salir?", "Qué horror", JOptionPane.***YES\_NO\_OPTION***);

**if** (respuesta == JOptionPane.***YES\_OPTION***){

**this**.dispose();

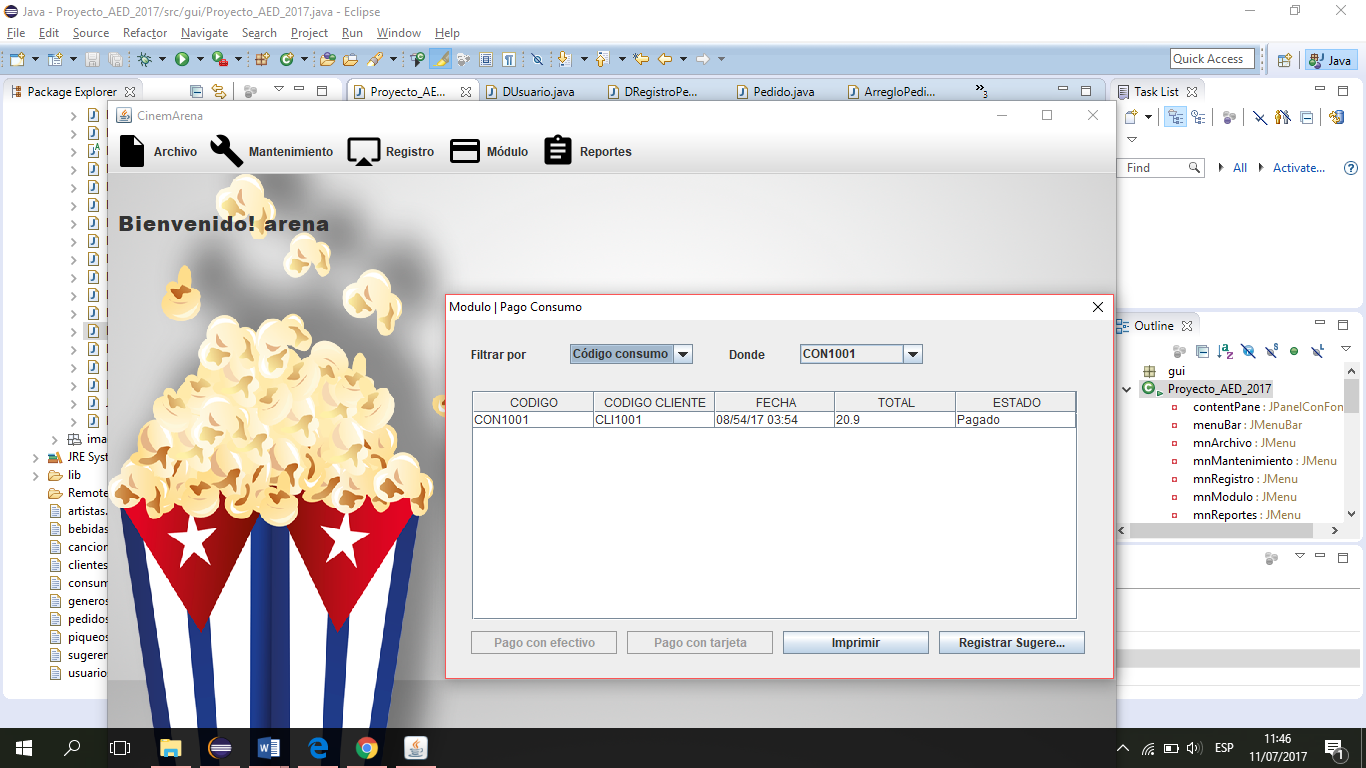
}

}

}

**MODULO**

* **MODULO PAGO CONSUMO**



**package** gui;

**import** java.awt.BorderLayout;

**import** java.awt.FlowLayout;

**import** javax.swing.DefaultComboBoxModel;

**import** javax.swing.JButton;

**import** javax.swing.JDialog;

**import** javax.swing.JPanel;

**import** javax.swing.border.EmptyBorder;

**import** javax.swing.table.DefaultTableModel;

**import** arreglo.ArregloCliente;

**import** arreglo.ArregloConsumo;

**import** arreglo.ArregloUsuario;

**import** clases.Cliente;

**import** clases.Consumo;

**import** clases.Genero;

**import** clases.Pedido;

**import** clases.Usuario;

**import** javax.swing.JTextField;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JComboBox;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTextArea;

**import** java.awt.event.ActionListener;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** java.util.ArrayList;

**import** java.awt.event.ActionEvent;

**import** javax.swing.JTable;

**public** **class** DPagoConsumo **extends** JDialog **implements** ActionListener, MouseListener {

**private** **static** **final** **int** ***CODCON***=0;

**private** **static** **final** **int** ***CODCLI***=1;

**private** **final** JPanel contentPanel = **new** JPanel();

**private** JLabel lblDonde;

**private** JComboBox<String> cboFiltrarPor;

**private** JComboBox<String> cboCodigo;

**private** JButton btnPagoEfectivo;

**private** JButton btnPagoTarjeta;

**private** JButton btnImprimir;

**private** JButton btnRegistrarSugerencia;

**private** JTable table;

**private** DefaultTableModel modelo = **new** DefaultTableModel();

**private** ArregloUsuario au = **new** ArregloUsuario();

**private** ArregloConsumo ac = **new** ArregloConsumo();

**private** ArregloCliente acl = **new** ArregloCliente();

**private** String[] idConsumos;

**private** String[] idClientes;

**private** **int** usuarioActual;

**private** **boolean** estado=**true**;

**private** ArrayList<Integer> idsSeleccionados;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

**try** {

DPagoConsumo dialog = **new** DPagoConsumo(1001);

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

dialog.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* Create the dialog.

\*/

**public** DPagoConsumo(**int** usuarioActual) {

**this**.usuarioActual = usuarioActual;

setTitle("Modulo | Pago Consumo ");

setBounds(100, 100, 673, 387);

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

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

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

contentPanel.setLayout(**null**);

JLabel lblFiltrarPor = **new** JLabel("Filtrar por");

lblFiltrarPor.setBounds(25, 27, 123, 14);

contentPanel.add(lblFiltrarPor);

cboFiltrarPor= **new** JComboBox<String>();

cboFiltrarPor.setBounds(124, 24, 123, 20);

cboFiltrarPor.setModel(**new** DefaultComboBoxModel<String>(**new** String[]{"Código consumo", "Código cliente"}));

cboFiltrarPor.addActionListener(**this**);

contentPanel.add(cboFiltrarPor);

cboCodigo = **new** JComboBox<String>();

cboCodigo.setBounds(354, 24, 123, 20);

cboCodigo.addActionListener(**this**);

contentPanel.add(cboCodigo);

btnPagoEfectivo = **new** JButton("Pago con efectivo");

btnPagoEfectivo.addActionListener(**this**);

btnPagoEfectivo.setBounds(25, 311, 146, 23);

btnPagoEfectivo.setEnabled(**false**);

contentPanel.add(btnPagoEfectivo);

btnPagoTarjeta = **new** JButton("Pago con tarjeta");

btnPagoTarjeta.addActionListener(**this**);

btnPagoTarjeta.setBounds(181, 311, 146, 23);

btnPagoTarjeta.setEnabled(**false**);

contentPanel.add(btnPagoTarjeta);

btnImprimir = **new** JButton("Imprimir");

btnImprimir.addActionListener(**this**);

btnImprimir.setBounds(337, 311, 146, 23);

contentPanel.add(btnImprimir);

btnRegistrarSugerencia = **new** JButton("Registrar Sugerencia ");

btnRegistrarSugerencia.addActionListener(**this**);

btnRegistrarSugerencia.setBounds(493, 311, 146, 23);

contentPanel.add(btnRegistrarSugerencia);

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(26, 71, 606, 229);

contentPanel.add(scrollPane);

modelo.addColumn("CODIGO");

modelo.addColumn("CODIGO CLIENTE");

modelo.addColumn("FECHA");

modelo.addColumn("TOTAL");

modelo.addColumn("ESTADO");

table = **new** JTable();

table.setFillsViewportHeight(**true**);

table.setModel(modelo);

table.addMouseListener(**this**);

scrollPane.setViewportView(table);

lblDonde = **new** JLabel("Donde");

lblDonde.setBounds(283, 27, 119, 14);

contentPanel.add(lblDonde);

llenarIds();

llenarCombos();

setResizable(**false**);

modelo.setRowCount(0);

llenarPorIdConsumo();

}

**private** **void** llenarTabla (**int** i){

Object[] datos ={ Consumo.***PRE***+ac.obtener(i).getCodConsumo(),

Cliente.***PRE***+acl.obtener(ac.obtener(i).getIdCliente()).getIdCliente(),

ac.obtener(i).getFecha(),

ac.obtener(i).getTotal(),

Consumo.*obtenerEstado*(ac.obtener(i).getEstado()),

};

modelo.addRow(datos);

}

**private** **void** llenarPorIdConsumo() {

idsSeleccionados = **new** ArrayList<Integer>();

**int** indiceConsumo = cboCodigo.getSelectedIndex();

idsSeleccionados.add(indiceConsumo);

modelo.setRowCount(0);

//Verifico que esté pendiente

**if** (ac.obtener(indiceConsumo).getEstado()==Consumo.***PEN***){

llenarTabla(indiceConsumo);

} **else** {

estado=**false**;

llenarTabla(indiceConsumo);

}

}

**private** **void** llenarPorIdCliente() {

**int** indiceCliente = cboCodigo.getSelectedIndex();

idsSeleccionados = **new** ArrayList<Integer>();

modelo.setRowCount(0);

//Simplemente obtengo los indices de clientes que tienen asociado un consumo

**for** (**int** i=0;i<ac.tamaño();i++){

**if** (ac.obtener(i).getIdCliente()==indiceCliente){

idsSeleccionados.add(i);

}

}

//Verifico que haya pedidos asociados. Si no, retorna

**if**(idsSeleccionados.size()>0){

**for** (**int** i=0;i<idsSeleccionados.size();i++){

llenarTabla(idsSeleccionados.get(i));

}

} **else** {

estado=**false**;

}

}

**private** **void** llenarIds() {

idConsumos = **new** String[ac.tamaño()];

**for**(**int** i=0;i<ac.tamaño();i++){

idConsumos[i] = Consumo.***PRE***+ac.obtener(i).getCodConsumo();

}

idClientes = **new** String[acl.tamaño()];

**for**(**int** i=0;i<acl.tamaño();i++){

idClientes[i] = Cliente.***PRE***+acl.obtener(i).getIdCliente();

}

}

**private** **void** llenarCombos() {

**int** indice = cboFiltrarPor.getSelectedIndex();

**switch** (indice){

**case** ***CODCON***:

cboCodigo.setModel(**new** DefaultComboBoxModel<String>(idConsumos));

cboCodigo.setSelectedIndex(0);

**return**;

**default**:

cboCodigo.setModel(**new** DefaultComboBoxModel<String>(idClientes));

cboCodigo.setSelectedIndex(0);

}

}

**public** **void** actionPerformed(ActionEvent arg0) {

**if**(arg0.getSource()==cboFiltrarPor){

llenarCombos();

}

**else** **if**(arg0.getSource()==cboCodigo){

estado=**true**;

**int** indice = cboFiltrarPor.getSelectedIndex();

**switch** (indice){

**case** ***CODCON***:

llenarPorIdConsumo();

**return**;

**default**:

llenarPorIdCliente();

}

}

**else** **if** (arg0.getSource() == btnImprimir) {

**int** indice = table.getSelectedRow();

**if** (indice>-1){

**int** indiceSeleccionado = idsSeleccionados.get(indice);

DBoleta dialog = **new** DBoleta();

dialog.escribir("");

dialog.escribir("Código consumo: "+Consumo.***PRE***+ac.obtener(indiceSeleccionado).getCodConsumo());

dialog.escribir("Cliente : "+acl.obtener(ac.obtener(indiceSeleccionado).getIdCliente()).getNombres()+" "+

acl.obtener(ac.obtener(indiceSeleccionado).getIdCliente()).getApellidoPaterno()+" "+

acl.obtener(ac.obtener(indiceSeleccionado).getIdCliente()).getApellidoMaterno());

dialog.escribir("Fecha : "+ac.obtener(indiceSeleccionado).getFecha());

dialog.escribir("Total : S/."+ac.obtener(indiceSeleccionado).getTotal());

dialog.escribir("Estado : "+Consumo.*obtenerEstado*(ac.obtener(indiceSeleccionado).getEstado()));

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

}

**else** **if** (arg0.getSource() == btnPagoTarjeta) {

**int** indice = table.getSelectedRow();

//Registro el pago

**if** (indice>-1){

**int** indiceSeleccionado = idsSeleccionados.get(indice);

Consumo consumo = ac.obtener(indiceSeleccionado);

consumo.setEstado(Consumo.***PAG***);

ac.reemplazar(consumo, indiceSeleccionado);

}

//Vuelvo a llenar la tabla con los estados actualizados

modelo.setRowCount(0);

**for** (**int** i=0;i<idsSeleccionados.size();i++){

llenarTabla(idsSeleccionados.get(i));

}

}

**else** **if** (arg0.getSource() == btnPagoEfectivo) {

**int** indice = table.getSelectedRow();

//Registro el pago

**if** (indice>-1){

**int** indiceSeleccionado = idsSeleccionados.get(indice);

Consumo consumo = ac.obtener(indiceSeleccionado);

consumo.setEstado(Consumo.***PAG***);

ac.reemplazar(consumo, indiceSeleccionado);

}

//Vuelvo a llenar la tabla con los estados actualizados

modelo.setRowCount(0);

**for** (**int** i=0;i<idsSeleccionados.size();i++){

llenarTabla(idsSeleccionados.get(i));

}

}

**else** {

**this**.dispose();

DSugerencia dialog = **new** DSugerencia(usuarioActual);

dialog.setLocationRelativeTo(**this**);

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

dialog.setModal(**true**);

dialog.setVisible(**true**);

}

}

@Override

**public** **void** mouseClicked(MouseEvent arg0) {

**int** indice = table.getSelectedRow();

**if** (indice>-1){

**int** indiceConsumo = idsSeleccionados.get(indice);

**if**(ac.obtener(indiceConsumo).getEstado()==Consumo.***PEN***){

btnPagoEfectivo.setEnabled(**true**);

btnPagoTarjeta.setEnabled(**true**);

} **else** {

btnPagoEfectivo.setEnabled(**false**);

btnPagoTarjeta.setEnabled(**false**);

}

}

}

@Override

**public** **void** mouseEntered(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseExited(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mousePressed(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

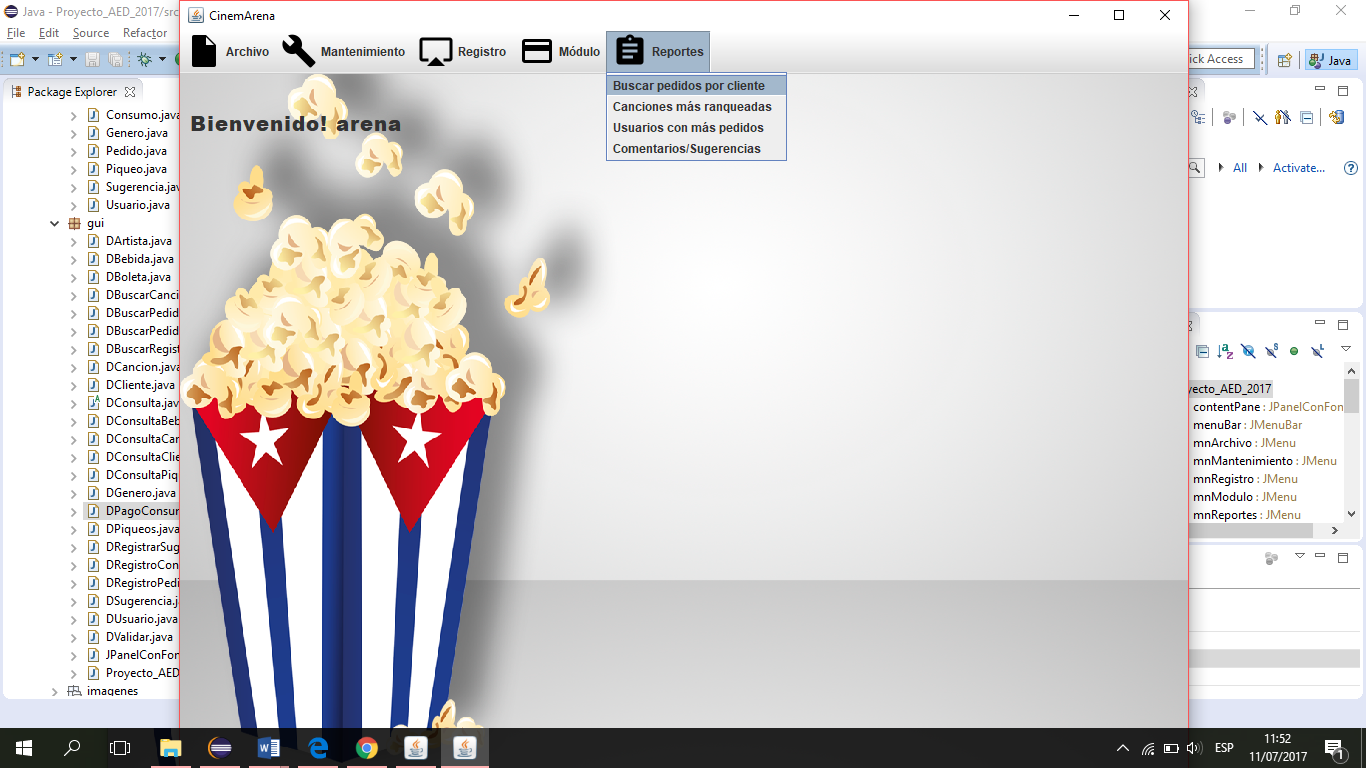
@Override

**public** **void** mouseReleased(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

}

**REPORTE** 

**GUI**

**DBuscarCancionesRanqueadas.java**

**package** gui;

**import** java.awt.BorderLayout;

**import** java.awt.FlowLayout;

**import** arreglo.ArregloArtista;

**import** arreglo.ArregloBebida;

**import** arreglo.ArregloCancion;

**import** clases.Bebida;

**import** clases.Cancion;

**import** javax.swing.JButton;

**import** javax.swing.JDialog;

**import** javax.swing.JPanel;

**import** javax.swing.border.EmptyBorder;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.text.html.CSS;

**import** java.awt.Window.Type;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JComboBox;

**import** javax.swing.DefaultComboBoxModel;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTextArea;

**import** javax.swing.JTextField;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** javax.swing.JTable;

**public** **class** DBuscarCancionesRanqueadas **extends** JDialog **implements** ActionListener {

**private** **static** **final** **int** CIN=0;

**private** **static** **final** **int** CUA=1;

**private** **static** **final** **int** TRE=2;

**private** **static** **final** **int** VEI=3;

**private** **static** **final** **int** DIE=4;

**private** JButton btnSalir;

**private** JComboBox<String> cboRanqueadas;

**private** JTable table;

**private** DefaultTableModel modelo = **new** DefaultTableModel();

**private** ArregloCancion ca = **new** ArregloCancion();

**private** ArregloArtista aa = **new** ArregloArtista();

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

**try** {

DBuscarCancionesRanqueadas dialog = **new** DBuscarCancionesRanqueadas();

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

dialog.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* Create the dialog.

\*/

**public** DBuscarCancionesRanqueadas(){

**if** (aa.tamaño()!=0){

setTitle("Reporte | Canciones más ranqueadas");

setBounds(100, 100, 439, 303);

getContentPane().setLayout(**null**);

JLabel lblCancionesRan = **new** JLabel("Top de las Canciones. Num. Reproducciones: ");

lblCancionesRan.setBounds(10, 11, 271, 14);

getContentPane().add(lblCancionesRan);

cboRanqueadas = **new** JComboBox<String>();

cboRanqueadas.setModel(**new** DefaultComboBoxModel<String>(**new** String[] {obtenerIndiceReproducciones(0),

obtenerIndiceReproducciones(1),obtenerIndiceReproducciones(2), obtenerIndiceReproducciones(3),

obtenerIndiceReproducciones(4)}));

cboRanqueadas.setBounds(316, 8, 108, 20);

cboRanqueadas.addActionListener(**this**);

cboRanqueadas.setSelectedIndex(0);

getContentPane().add(cboRanqueadas);

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(10, 34, 414, 179);

getContentPane().add(scrollPane);

table = **new** JTable();

table.setFillsViewportHeight(**true**);

table.setModel(modelo);

scrollPane.setViewportView(table);

modelo.addColumn("CODIGO");

modelo.addColumn("NOMBRE");

modelo.addColumn("DURACION");

modelo.addColumn("ARTISTA");

modelo.addColumn("ALBUM");

modelo.addColumn("REPRODUCCIONES");

modelo.addColumn("ESTADO");

btnSalir = **new** JButton("Salir");

btnSalir.setBounds(177, 227, 89, 23);

btnSalir.addActionListener(**this**);

getContentPane().add(btnSalir);

modelo.setRowCount(0);

llenarTablaConLimite(50);

setResizable(**false**);

setLocationRelativeTo(**this**);

setDefaultCloseOperation(JDialog.DISPOSE\_ON\_CLOSE);

setModal(**true**);

setVisible(**true**);

} **else** {

**this**.dispose();

JOptionPane.showMessageDialog(**this**, "Necesita registrar al menos un artista", "Cuidado, mucho cuidado",

JOptionPane.ERROR\_MESSAGE);

}

}

**private** **void** llenarTabla (){

**for**(**int** i=0 ; i<ca.tamaño(); i++){

Object[] datos ={ Cancion.PRE+ca.obtener(i).getidCancion(),

ca.obtener(i).getnombre(),

ca.obtener(i).getduracion(),

aa.obtener(ca.obtener(i).getidArtista()).getNombreArtistico(),

ca.obtener(i).getalbum(),

ca.obtener(i).getnumeroReproducciones(),

Cancion.obtenerEstado(ca.obtener(i).getestado()),

};

modelo.addRow(datos);

}

}

**private** **void** llenarTablaConLimite (**int** limite){

**for**(**int** i=0 ; i<ca.tamaño(); i++){

Object[] datos ={ Cancion.PRE+ca.obtener(i).getidCancion(),

ca.obtener(i).getnombre(),

ca.obtener(i).getduracion(),

aa.obtener(ca.obtener(i).getidArtista()).getNombreArtistico(),

ca.obtener(i).getalbum(),

ca.obtener(i).getnumeroReproducciones(),

Cancion.obtenerEstado(ca.obtener(i).getestado()),

};

//Solo se mostrarán los que tengan el limite

**if**(ca.obtener(i).getnumeroReproducciones()>=limite){

System.out.println(ca.obtener(i).getnumeroReproducciones());

modelo.addRow(datos);

}

}

}

**private** String obtenerIndiceReproducciones(**int** caso){

**switch**(caso){

**case** CIN:

**return** "50";

**case** CUA:

**return** "40";

**case** TRE:

**return** "30";

**case** VEI:

**return** "20";

**default**:

**return** "10";

}

}

@Override

**public** **void** actionPerformed(ActionEvent arg0) {

**if**(arg0.getSource()==cboRanqueadas){

**int** indiceAcual = cboRanqueadas.getSelectedIndex();

**int** limiteActual = Integer.parseInt(obtenerIndiceReproducciones(indiceAcual));

modelo.setRowCount(0);

llenarTablaConLimite(limiteActual);

}

**else** {

**this**.dispose();

}

}

}