**GA7-220501096-AA3-EV01**

**codificación de módulos del software Stand alone, web y móvil DE ACUERDO CON EL proyecto a desarrollar**

**ACTIVIDAD DE APRENDIZAJE**

**FASE DE EJECUCIÓN**

**FICHA 2721478**

**Aprendiz:**

**WALTER SEBASTIAN MUÑOZ ALFARO**

**SERVICIO NACIONAL DE APRENDIZAJE - SENA**

**TECNÓLOGO ANALISIS Y DESARROLLO DE SOFTWARE**

**PROYECTO FORMACIÓN TITULADA VIRTUAL**

**2024**

**Introducción**

El propósito de este proyecto es desarrollar una robusta aplicación CRUD (Crear, Leer, Actualizar y Eliminar) para gestionar datos de estudiantes de manera eficiente y confiable. Este sistema está construido utilizando el lenguaje de programación Java, conocido por  su versatilidad y escalabilidad, y se apoya en una base de datos MySQL, la cual asegura un almacenamiento seguro y una recuperación rápida de la información.

En el entorno educativo y administrativo actual, contar con un sistema de gestión de datos eficiente no es solo una ventaja, sino una necesidad. La implementación de esta aplicación facilita el acceso a datos precisos y actualizados, lo cual es vital para la toma de decisiones informadas. Además, mejora significativamente la optimización de los recursos dentro de las instituciones educativas, permitiendo a los administradores y profesores enfocarse en mejorar la calidad educativa en lugar de en tareas administrativas repetitivas.

**Objetivo general**

Desarrollar una aplicación de gestión de estudiantes en Java con MySQL que permita a los usuarios realizar operaciones CRUD de manera intuitiva y eficiente.

**Objetivos específicos**

* Conecte la aplicación Java con una base de datos MySQL utilizando JDBC.
* Implementar las operaciones CRUD para gestionar la información de los estudiantes.
* Desarrollar una interfaz gráfica amigable en Java para interactuar con los datos.
* Facilitar la modificación y eliminación de datos en la base de datos desde la aplicación.

**Tecnologías utilizadas**

* Java: Lenguaje principal del desarrollo.
* MySQL: Sistema de gestión de bases de datos utilizadas para almacenar la información.
* JDBC (Java Database Connectivity): Biblioteca para la conexión entre Java y MySQL.
* NetBeans IDE: Entorno de desarrollo integrado que facilita la gestión de proyectos en Java.

**Estructura de clases**

El proyecto consta de las siguientes clases principales:

* Clase CAlumnos: Esta clase contiene las operaciones CRUD para los registros de estudiantes.

Métodos principales: InsertarAlumno, MostrarAlumnos, SeleccionarAlumno, ModificarAlumnos, EliminarAlumnos.

* Clase CConexion: Gestiona la conexión entre la aplicación y la base de datos MySQL. Defina los parámetros de conexión y maneje posibles errores de conexión.
* Clase inicio: Es la clase principal que lanza la interfaz gráfica de la aplicación.
* Clase interfazAlumno: Define y gestiona los elementos gráficos (JTable, JTextField, JButton, etc.) y las interacciones con los usuarios.

**Funcionalidades implementadas**

* Insertar Alumnos:  
  Permite agregar un nuevo alumno en la base de datos. Al llenar los campos de nombre y apellido, se llama al método InsertarAlumnoque envía los datos al servidor MySQL.
* Mostrar Alumnos:  
  Muestra en una tabla todos los registros de alumnos almacenados en la base de datos. Esta función utiliza MostrarAlumnos, que recupera los datos a través de una consulta SQL.
* Seleccionar Alumnos:  
  Permite seleccionar un registro en la tabla de alumnos y cargar sus datos en los campos de texto correspondientes. Esto facilita la modificación o eliminación de registros.
* Modificar Alumnos:  
  Esta funcionalidad permite cambiar los datos de un alumno seleccionado. Los campos de nombre y apellido pueden actualizarse y el cambio se reflejará en la base de datos al ejecutar ModificarAlumnos.
* Eliminar Alumnos:  
  Proporciona la capacidad de eliminar un registro de alumno seleccionado en la tabla. Este proceso es ejecutado por el método EliminarAlumnos, eliminando permanentemente el registro en la base de datos.

**Conclusiones**

La implementación de este proyecto ha permitido desarrollar una aplicación CRUD en Java, conectada con MySQL, útil en la gestión de datos de estudiantes. La aplicación cumple con los objetivos planteados al inicio y demuestra la utilidad de una interfaz gráfica en la gestión y administración de datos.

La aplicación también se puede expandir para soportar nuevas funcionalidades, como la búsqueda avanzada de registros, la adición de roles para usuarios, o incluso la migración a otras tecnologías de bases de datos. Con esta base, se han establecido las herramientas necesarias para una futura escalabilidad.

Anexos (Código Fuente)

ClaseCAlumnos

[/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package walter.java\_crud\_mysql;

// Importación de clases necesarias para interfaces gráficas y SQL

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import java.sql.CallableStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

import javax.swing.table.TableRowSorter;

import javax.swing.JTable;

public class CAlumnos {

// Atributos de la clase CAlumnos

int codigo;

String nombreAlumnos;

String apelidoAlumnos;

// Getters y Setters

public int getCodigo() {

return codigo;

}

public void setCodigo(int codigo) {

this.codigo = codigo;

}

public String getNombreAlumnos() {

return nombreAlumnos;

}

public void setNombreAlumnos(String nombreAlumnos) {

this.nombreAlumnos = nombreAlumnos;

}

public String getApelidoAlumnos() {

return apelidoAlumnos;

}

public void setApelidoAlumnos(String apelidoAlumnos) {

this.apelidoAlumnos = apelidoAlumnos;

}

// Método para insertar un nuevo alumno en la base de datos

public void InsertarAlumno(JTextField paramNombres, JTextField paramApellidos) {

setNombreAlumnos(paramNombres.getText());

setApelidoAlumnos(paramApellidos.getText());

CConexion objetoConexion = new CConexion();

String consulta = "insert into Alumnos (nombres, apellidos) values(?, ?);";

try {

CallableStatement cs = objetoConexion.estableceConexion().prepareCall(consulta);

cs.setString(1, getNombreAlumnos());

cs.setString(2, getApelidoAlumnos());

cs.execute();

JOptionPane.showMessageDialog(null, "Se inserto correctamente el alumno");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, "No se inserto correctamente el alumno, error: " + e.toString());

}

}

// Método para mostrar los alumnos en una tabla

public void MostrarAlumnos(JTable paramTablaTotalAlumnos){

CConexion objetoConexion = new CConexion();

DefaultTableModel modelo = new DefaultTableModel();

TableRowSorter<TableModel> OrdenarTabla = new TableRowSorter<TableModel>(modelo);

paramTablaTotalAlumnos.setRowSorter(OrdenarTabla);

String sql="";

// Definición de columnas en el modelo

modelo.addColumn("Id");

modelo.addColumn("Nombres");

modelo.addColumn("apellidos");

paramTablaTotalAlumnos.setModel (modelo);

sql = "select \* from Alumnos;";

String[] datos = new String[3];

Statement st;

try {

st= objetoConexion.estableceConexion().createStatement();

ResultSet rs = st.executeQuery(sql);

// Llenado de datos en la tabla

while (rs.next()) {

datos[0]=rs.getString(1);

datos[1]=rs.getString(2);

datos[2]=rs.getString(3);

modelo.addRow(datos);

}

paramTablaTotalAlumnos.setModel (modelo);

} catch (Exception e) {

JOptionPane.showMessageDialog(null, "No se pudo mostrsr los registros, error: "+e.toString());

}

}

// Método para seleccionar un alumno de la tabla y mostrarlo en campos de texto

public static void SeleccionarAlumno(JTable paramTablaAlumnos, JTextField paramId, JTextField paramNombres, JTextField paramApellidos)

{

try {

int fila = paramTablaAlumnos.getSelectedRow();

if (fila >=0){

paramId.setText((String) (paramTablaAlumnos.getValueAt(fila, 0)));

paramNombres.setText((String) (paramTablaAlumnos.getValueAt(fila, 1)));

paramApellidos.setText((String) (paramTablaAlumnos.getValueAt(fila, 2)));

}

else

{

JOptionPane.showMessageDialog(null, "Fila no seleccionada");

}

} catch (Exception e) {

JOptionPane.showMessageDialog(null, "Error de seleccion, error: "+e.toString());

}

}

// Método para modificar datos de un alumno

public void ModificarAlumnos (JTextField paramCodigo, JTextField paramNombres, JTextField paramApellidos){

setCodigo(Integer.parseInt(paramCodigo.getText()));

setNombreAlumnos(paramNombres.getText());

setApelidoAlumnos(paramApellidos.getText());

CConexion objeConexion = new CConexion();

String consulta = "update Alumnos set alumnos.nombres =?, alumnos.apellidos =? where alumnos.id=?";

try {

CallableStatement cs = objeConexion.estableceConexion().prepareCall(consulta);

cs.setString(1, getNombreAlumnos());

cs.setString(2, getApelidoAlumnos());

cs.setInt(3, getCodigo());

cs.execute();

JOptionPane.showMessageDialog(null, "modificacion exitosa.");

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, "No se modifico, error: "+e.toString());

}

}

// Método para eliminar un alumno

public void EliminarAlumnos(JTextField paramCodigo){

setCodigo(Integer.parseInt(paramCodigo.getText()));

CConexion objetoConexion = new CConexion();

String consulta = "DELETE FROM Alumnos where alumnos.id=?;";

try {

CallableStatement cs = objetoConexion.estableceConexion().prepareCall(consulta);

cs.setInt(1, getCodigo());

cs.execute();

JOptionPane.showMessageDialog(null, "Se ha eliminado correctamente.");

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, "No se elimino, error: "+e.toString());

}

}

}]

ClaseCConexion

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package walter.java\_crud\_mysql;

import java.sql.Connection;

import java.sql.DriverManager;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author ASUS

\*/

public class CConexion {

// Variables para conexión a la base de datos

Connection conectar = null;

String usuario ="root";

String contraseña ="1929ana";

String bd ="bdescuela";

String ip ="localhost";

String puerto ="3306";

String cadena = "jdbc:mysql://"+ip+":"+puerto+"/"+bd;

// Método para establecer la conexión con la base de datos

public Connection estableceConexion (){

try {

Class.forName("com.mysql.cj.jdbc.Driver");

conectar = DriverManager.getConnection(cadena,usuario,contraseña);

//JOptionPane.showMessageDialog(null, "La conexion se ha realizado con exito");

} catch (Exception e){

JOptionPane.showMessageDialog(null,"Error al conectarse ala base de datos, error: "+e.toString());

}

return conectar;

}

} []

Claseinicio

[]/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package walter.java\_crud\_mysql;

/\*\*

\*

\* @author ASUS

\*/

public class inicio {

// Método principal para iniciar la interfaz gráfica

public static void main(String[] args) {

interfazAlumno objetoFormulario = new interfazAlumno();

objetoFormulario.setVisible(true);

}

}

ClaseinterfazAlumno

[]/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template

\*/

package walter.java\_crud\_mysql;

/\*\*

\*

\* @author ASUS

\*/

public class interfazAlumno extends javax.swing.JFrame {

/\*\*

\* Creates new form interfazAlumno

\*/

public interfazAlumno() {

initComponents();

this.setLocationRelativeTo(null);

txtId.setEditable(false);

// CConexion objetoConexion = new CConexion ();

// objetoConexion.estableceConexion();

CAlumnos objetoAlumnos= new CAlumnos();

objetoAlumnos.MostrarAlumnos(tbTotalAlumnos);

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

txtId = new javax.swing.JTextField();

txtNombres = new javax.swing.JTextField();

txtApellidos = new javax.swing.JTextField();

btnGuardar = new javax.swing.JButton();

btnModificar = new javax.swing.JButton();

btnEliminar = new javax.swing.JButton();

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

tbTotalAlumnos = new javax.swing.JTable();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder("datos alumno"));

jLabel1.setText("Id:");

jLabel2.setText("Nombres:");

jLabel3.setText("Apellidos:");

txtId.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

txtIdActionPerformed(evt);

}

});

txtNombres.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

txtNombresActionPerformed(evt);

}

});

btnGuardar.setText("guardar");

btnGuardar.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnGuardarActionPerformed(evt);

}

});

btnModificar.setText("modificar");

btnModificar.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnModificarActionPerformed(evt);

}

});

btnEliminar.setText("eliminar");

btnEliminar.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnEliminarActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

.addComponent(btnEliminar, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(btnModificar, javax.swing.GroupLayout.DEFAULT\_SIZE, 102, Short.MAX\_VALUE)

.addComponent(btnGuardar, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 37, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 57, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 75, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 18, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(txtId, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 6, Short.MAX\_VALUE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(txtApellidos)

.addComponent(txtNombres))

.addGap(11, 11, 11))))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(47, 47, 47)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(txtId, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(28, 28, 28)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(txtNombres, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(21, 21, 21)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(txtApellidos, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(56, 56, 56)

.addComponent(btnGuardar)

.addGap(18, 18, 18)

.addComponent(btnModificar)

.addGap(18, 18, 18)

.addComponent(btnEliminar)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jPanel2.setBorder(javax.swing.BorderFactory.createTitledBorder("lista de alumnos"));

tbTotalAlumnos.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

{},

{},

{},

{}

},

new String [] {

}

));

tbTotalAlumnos.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

tbTotalAlumnosMouseClicked(evt);

}

});

jScrollPane1.setViewportView(tbTotalAlumnos);

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(25, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap(13, Short.MAX\_VALUE)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(41, 41, 41))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(34, 34, 34)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addContainerGap(17, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void txtNombresActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {

CAlumnos objetoAlumno = new CAlumnos();

objetoAlumno.InsertarAlumno(txtNombres, txtApellidos);

objetoAlumno.MostrarAlumnos(tbTotalAlumnos);

}

private void tbTotalAlumnosMouseClicked(java.awt.event.MouseEvent evt) {

CAlumnos objetoAlumno = new CAlumnos();

objetoAlumno.SeleccionarAlumno(tbTotalAlumnos, txtId, txtNombres, txtApellidos);

}

private void txtIdActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void btnModificarActionPerformed(java.awt.event.ActionEvent evt) {

CAlumnos objetoAlumno = new CAlumnos();

objetoAlumno.ModificarAlumnos(txtId,txtNombres, txtApellidos);

objetoAlumno.MostrarAlumnos(tbTotalAlumnos);

}

private void btnEliminarActionPerformed(java.awt.event.ActionEvent evt) {

CAlumnos objetoAlumno = new CAlumnos();

objetoAlumno.EliminarAlumnos(txtId);

objetoAlumno.MostrarAlumnos(tbTotalAlumnos);

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(interfazAlumno.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(interfazAlumno.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(interfazAlumno.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(interfazAlumno.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new interfazAlumno().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnEliminar;

private javax.swing.JButton btnGuardar;

private javax.swing.JButton btnModificar;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable tbTotalAlumnos;

private javax.swing.JTextField txtApellidos;

private javax.swing.JTextField txtId;

private javax.swing.JTextField txtNombres;

// End of variables declaration

}