



TECNOLOGICO NACIONAL DE MEXICO

Instituto Tecnológico de la Laguna
Ingeniería en Sistemas Computacionales



TOPICOS AVANZADOS DE PROGRAMACION

PERIODO: Ago - Dic/ 2020

GRUPO: "B" 17 – 18 Hrs

PRACTICA No. U3P01

Aplicación de Base de Datos con JavaDB

ALUMNO:

17130800 Félix Gerardo Martínez Hinojo

PROFESOR:

Ing. Luis Fernando Gil Vázquez

Torreón, Coah. A 07 de Enero de 2021

Situación didáctica

En una escuela secundaria local se está buscando a desarrolladores de software que puedan programar una aplicación de base de datos en Java para el control de calificaciones de los alumnos. La primera versión se espera que cuente con tres funciones principales:

- Gestión de un catálogo de alumnos.
- Gestión de un catálogo de materias.
- Gestión del Kardex de calificación de alumnos.

Derivado de un levantamiento de requerimientos preliminar se ha determinado lo siguiente:

1. La gestión de cada catálogo debe contar con las operaciones **CRUD** (Create, Update, Delete), también conocidas como ABC (Altas-Bajas-Cambios).
2. Para el catálogo de alumnos requiere: matrícula, nombre del alumno, apellidos, promedio
3. Para catálogo de materias se requiere: clave de la materia, materia y créditos.
4. Para el Kardex de calificaciones se requiere: matrícula del alumno, materia, periodo, calificación.
5. Se requiere que el sistema proporcione opciones de consulta para filtrar datos que cumplan un cierto criterio de selección.
6. Se requiere que el sistema cuente con una función de recalcado de promedios.

Como una restricción la escuela solicita que la base de datos del sistema use **JavaDB** como Sistema Manejador de Base de Datos (DBMS).

En la sección de **ANÁLISIS** presentar la siguiente información:

- Información de conexión a la base de datos:
 - Nombre del controlador JDBC
 - Cadena de conexión
 - Nombre de la base de datos
 - Usuario y contraseña
- Presentar un cuadro con la siguiente información de las tablas a crear en la base de datos:
 - Nombre de la tabla
 - Por cada columna:
 - Nombre de la columna
 - Tipo de dato
 - Es llave primaria
 - Es llave foránea

En la sección de **DISEÑO** va el diagrama de clases UML de la aplicación.

En la sección de **CODIGO** presentar el código del script de creación de la base de datos, enseguida el código de todas las clases JAVA de la aplicación con la calidad establecida.

En la sección de **PRUEBA DE EJECUCION** presentar un mini manual de usuario incluyendo capturas de pantalla de la aplicación.

Sección adicional: **ANEXOS**. Colocar aquí el script para poblar de registros las tablas de la base de datos de manera inicial.

En la entrega de esta práctica el archivo RAR debe incluir lo siguiente:

1. Este documento en PDF
2. Código fuente de la aplicación
3. Scripts de creación de la base de datos y de población inicial de tablas.

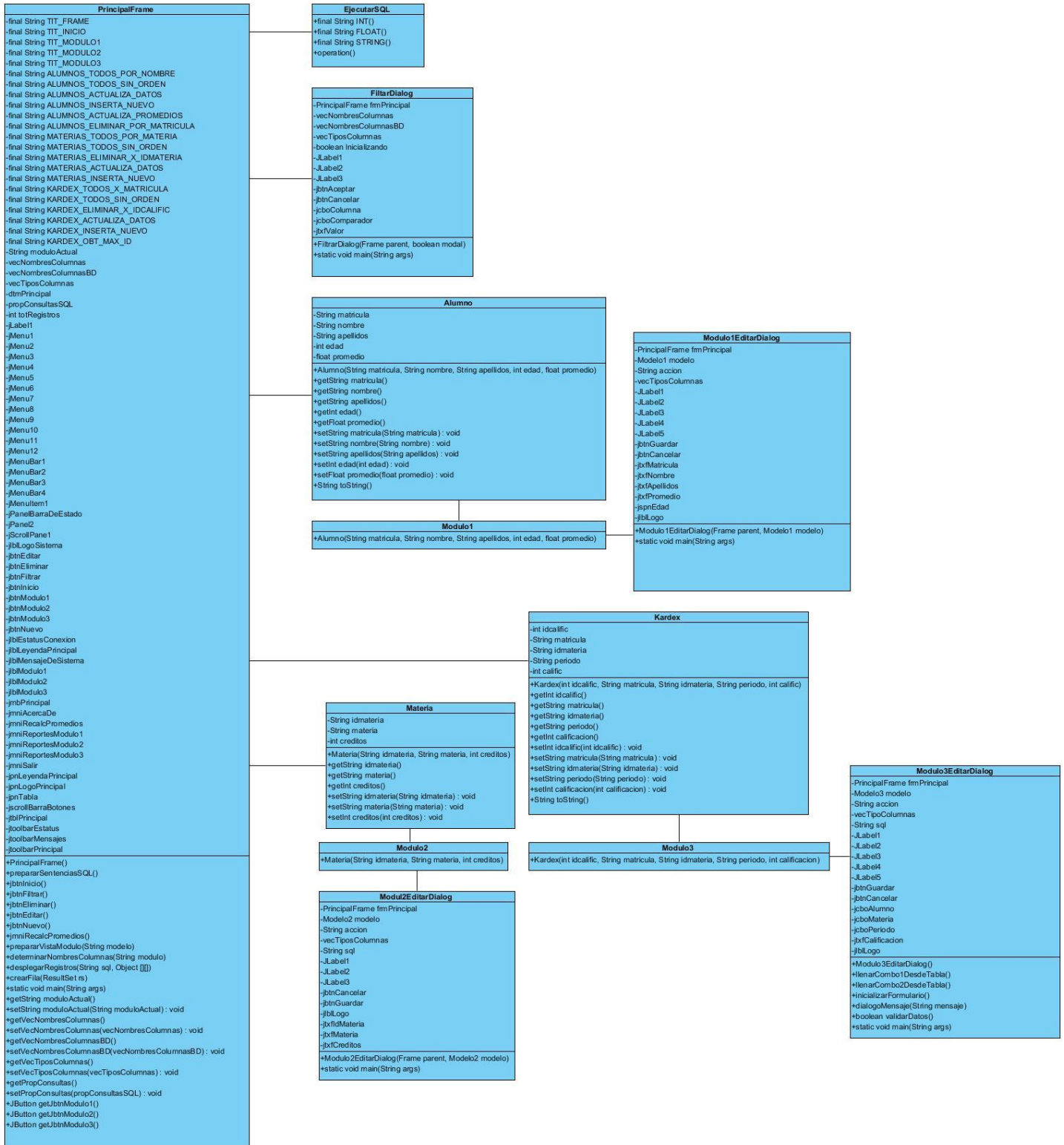
Análisis

- Información de conexión a la base de datos:
 - Nombre de la base de datos: **TAPdb**
 - Nombre De Usuario: **topicos**
 - Contraseña: **2020**

Nombre	Columnas
Alumnos	MATRICULA VARCHAR(10) primary key, NOMBRE VARCHAR(30) not null, APELLIDOS VARCHAR(50) not null, EDAD INTEGER, PROMEDIO REAL
Materias	IDMATERIA VARCHAR(10) primary key,, MATERIA VARCHAR(50) not null, CREDITOS INTEGER
Kardex	IDCALIF INTEGER primary key, MATRICULA VARCHAR(10), IDMATERIA VARCHAR(10), PERIODO VARCHAR(5), CALIFICACION INTEGER fk_matricula foreign key fk_idmateria foreign key

Diseño

Diseño de Clases:



Código

script-creación tablas en bd TAPdb.sql

```

/*-----
*:
*:          TECNOLOGICO NACIONAL DE MEXICO
*:          INSTITUTO TECNOLOGICO DE LA LAGUNA
*:          INGENIERIA EN SISTEMAS COMPUTACIONALES
*:          TOPICOS AVANZADOS DE PROGRAMACION "B"
*:
*:          SEMESTRE: AGO-DIC/2020      HORA: 17-18 HRS
*:
*:          Script de Creación de Tablas de la Base de Datos TAPdb
*:
*: Archivo      : script-creacion tablas en bd TAPdb.sql
*: Autor        : Félix Gerardo Martínez Hinojo      17130800
*: Fecha        : 03/Dic/2020
*: Compilador   :
*: Descripción  : Script SQL para crear las tablas Alumnos, Materias y Kardex
*:                y poblarlas con registros de prueba.
*: Última modif:
*: Fecha        Modificó          Motivo
*:=====
*: 03/Dic/2020 Félix Martínez      Se agrego el prologo
*:-----*/

/* CREACIÓN DE LAS TABLAS DE LA BASE DE DATOS */
CREATE TABLE Alumnos (
  MATRICULA VARCHAR ( 10 ) PRIMARY KEY,
  NOMBRE VARCHAR ( 30 ) NOT NULL,
  APELLIDOS VARCHAR ( 50 ) NOT NULL,
  EDAD INTEGER,
  PROMEDIO REAL
);

CREATE TABLE Materias (
  IDMATERIA VARCHAR ( 10 ) PRIMARY KEY,
  MATERIA VARCHAR ( 50 ) NOT NULL,
  CREDITOS INTEGER
);

CREATE TABLE Kardex (
  IDCALIFIC INTEGER PRIMARY KEY,
  MATRICULA VARCHAR ( 10 ),
  IDMATERIA VARCHAR ( 10 ),
  PERIODO VARCHAR ( 5 ),
  CALIFICACION INTEGER
);

/* CREACIÓN DE LLAVES FORANEAS PARA LA INTEGRIDAD REFERENCIAL */

ALTER TABLE Kardex ADD CONSTRAINT fk_matricula
  FOREIGN KEY ( MATRICULA ) REFERENCES Alumnos ( MATRICULA );

ALTER TABLE Kardex ADD CONSTRAINT fk_idmateria
  FOREIGN KEY ( IDMATERIA ) REFERENCES Materias ( IDMATERIA );

```

Alumno.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Clase que permite dar formato a todo lo relacionado con el Alumno.
.*
.* Archivo   : Alumno.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información de los Alumnos tales como:
.*
.*          - Matricula
.*          - Nombre
.*          - Apellidos
.*          - Edad
.*          - Promedio
.*
.* Ultima modif:
.* Fecha      Modific*      Motivo
.*-----
.* 07/DIC/2020 Félix Mtz   Se añadió la clase al proyecto y se le dió la definición básica..
.* 07/DIC/2020 Félix Mtz   Se añadieron los metodos pertinentes para acceder a la información.
.*-----*/
package modelo;

```

```

public class Alumno {

    private String matricula;
    private String nombre;
    private String apellidos;
    private int edad;
    private float promedio;

    // -----
    public Alumno ( String matricula, String nombre, String apellidos, int edad, float promedio ) {
        this.matricula = matricula;
        this.nombre = nombre;
        this.apellidos = apellidos;
        this.edad = edad;
        this.promedio = promedio;
    }
    // -----
    public String getMatricula() {
        return matricula;
    }
    // -----
    public void setMatricula ( String matricula ) {
        this.matricula = matricula;
    }
    // -----
    public String getNombre() {
        return nombre;
    }

```

```
}  
// -----  
public void setNombre ( String nombre ) {  
    this.nombre = nombre;  
}  
// -----  
public String getApellidos() {  
    return apellidos;  
}  
// -----  
public void setApellidos ( String apellidos ) {  
    this.apellidos = apellidos;  
}  
// -----  
public int getEdad() {  
    return edad;  
}  
// -----  
public void setEdad (int edad ) {  
    this.edad = edad;  
}  
// -----  
public float getPromedio() {  
    return promedio;  
}  
// -----  
public void setPromedio ( float promedio ) {  
    this.promedio = promedio;  
}  
// -----  
  
@Override  
public String toString () {  
    return nombre + " " + apellidos;  
}  
  
}
```

Kardex.java

```

/*-----
.*
.*      TECNOLOGICO NACIONAL DE MEXICO
.*      INSTITUTO TECNOLOGICO DE LA LAGUNA
.*      INGENIERIA EN SISTEMAS COMPUTACIONALES
.*      TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*      SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*      Clase que permite dar formato a todo lo relacionado con el Kardex del Alumno.
.*
.* Archivo   : Kardex.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información de los Alumnos A traves del Kardex:
.*
.*           - Matricula
.*           - Nombre
.*           - Apellidos
.*           - Edad
.*           - Promedio
.*           - Entre otras como las materias pertenecientes al alumno y demás.
.* Última modif:
.* Fecha      Modific♦      Motivo
.*=====
.* 07/DIC/2020 Félix Mtz    Se añadió la clase al proyecto y se le dió la definición básica..
.* 07/DIC/2020 Félix Mtz    Se añadieron los metodos pertinentes para acceder a la información.
.*-----*/
package modelo;

public class Kardex {

    private int    idcalific;
    private String matricula;
    private String idmateria;
    private String periodo;
    private int    calificacion;

    // -----
    public Kardex ( int idcalific, String matricula, String idmateria, String periodo, int calificacion ) {
        this.idcalific = idcalific;
        this.matricula = matricula;
        this.idmateria = idmateria;
        this.periodo = periodo;
        this.calificacion = calificacion;
    }
    // -----
    public int getIdcalific() {
        return idcalific;
    }
    // -----
    public void setIdcalific ( int idcalific ) {
        this.idcalific = idcalific;
    }
    // -----
    public String getMatricula() {
        return matricula;
    }

```



```
}  
// -----  
public void setMatricula ( String matricula ) {  
    this.matricula = matricula;  
}  
// -----  
public String getIdmateria() {  
    return idmateria;  
}  
// -----  
public void setIdmateria ( String idmateria ) {  
    this.idmateria = idmateria;  
}  
// -----  
public String getPeriodo() {  
    return periodo;  
}  
// -----  
public void setPeriodo ( String periodo ) {  
    this.periodo = periodo;  
}  
// -----  
public int getCalificacion() {  
    return calificacion;  
}  
// -----  
public void setCalificacion ( int calificacion ) {  
    this.calificacion = calificacion;  
}  
  
}
```

Materia.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Clase que permite dar formato a todo lo relacionado con la materia.
.*
.* Archivo   : Materia.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información de las materias tales como:
.*
.*          - ID Materia
.*          - Materia
.*          - Creditos
.*
.* Ultima modif:
.* Fecha      Modificó      Motivo
.*-----
=
.* 07/DIC/2020 Félix Mtz   Se añadió la clase al proyecto y se le dió la definición básica..
.* 07/DIC/2020 Félix Mtz   Se añadieron los metodos pertinentes para acceder a la información.
.*-----*/
package modelo;

public class Materia {

    private String idmateria;
    private String materia;
    private int    creditos;

    // -----
    public Materia ( String idmateria, String materia, int creditos ) {
        this.idmateria = idmateria;
        this.materia = materia;
        this.creditos = creditos;
    }
    // -----
    public String getIdmateria() {
        return idmateria;
    }
    // -----
    public void setIdmateria ( String idmateria ) {
        this.idmateria = idmateria;
    }
    // -----
    public String getMateria() {
        return materia;
    }
    // -----
    public void setMateria ( String materia ) {
        this.materia = materia;
    }

```

```
}  
// -----  
public int getCreditos() {  
    return creditos;  
}  
// -----  
public void setCreditos ( int creditos ) {  
    this.creditos = creditos;  
}  
  
// -----  
@Override  
public String toString () {  
    return materia + "";  
}  
  
}
```

Modelo1.java

```

/*-----
.*
.*      TECNOLOGICO NACIONAL DE MEXICO
.*      INSTITUTO TECNOLOGICO DE LA LAGUNA
.*      INGENIERIA EN SISTEMAS COMPUTACIONALES
.*      TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*
.*      SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*      Clase Modelo para Alumnos
.*
.* Archivo   : Modelo1.java
.* Autor    : Félix Gerardo Martínez Hinojo   17130800
.* Fecha    : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información de los Alumnos, el modelo de datos de los Alumnos
.*
.*          - Matricula
.*          - Nombre
.*          - Apellidos
.*          - Edad
.*          - Promedio
.*
.* Última modif:
.* Fecha      Modificó      Motivo
.*=====
.* 07/DIC/2020 Félix Mtz    Se añadió la clase al proyecto y se le dió la definición básica..
.*-----*/
package modelo;

public class Modelo1 extends Alumno {

    public Modelo1 ( String matricula, String nombre, String apellidos, int edad, float promedio ) {
        super ( matricula, nombre, apellidos, edad, promedio );
    }

}

```

Modelo2.java

```

/*-----
.*
.*      TECNOLOGICO NACIONAL DE MEXICO
.*      INSTITUTO TECNOLOGICO DE LA LAGUNA
.*      INGENIERIA EN SISTEMAS COMPUTACIONALES
.*      TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*
.*      SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*      Clase Modelo para Materias
.*
.* Archivo   : Modelo2.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información de las Materias, el modelo de datos de las Materias
.*
.*      - ID Matricula
.*      - Materia
.*      - Creditos
.*
.* Última modif:
.* Fecha      Modific♦      Motivo
.*=====
.* 07/DIC/2020 Félix Mtz    Se añadió la clase al proyecto y se le dió la definición básica..
.*-----*/
package modelo;

public class Modelo2 extends Materia {

    public Modelo2 ( String idmateria, String materia, int credits ) {
        super ( idmateria, materia, credits );
    }

}

```

Modelo3.java

```

/*-----
.*
.*      TECNOLOGICO NACIONAL DE MEXICO
.*      INSTITUTO TECNOLOGICO DE LA LAGUNA
.*      INGENIERIA EN SISTEMAS COMPUTACIONALES
.*      TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*
.*      SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*      Clase Modelo para Kardex
.*
.*
.* Archivo   : Modelo3.java
.* Autor    : Félix Gerardo Martínez Hinojo   17130800
.* Fecha    : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase que permite manejar la información del Kardex, el modelo de datos del Kardex
.*
.*
.* Última modif:
.* Fecha      Modific♦      Motivo
.*=====
.* 07/DIC/2020 Félix Mtz    Se añadió la clase al proyecto y se le dió la definición básica..
.*-----*/
package modelo;

public class Modelo3 extends Kardex {

    public Modelo3 ( int idcalific, String matricula, String idmateria, String periodo, int calificacion ) {
        super ( idcalific, matricula, idmateria, periodo, calificacion );
    }

}

```

EjecutorSQL.java

```

/*-----
*:
*:          TECNOLOGICO NACIONAL DE MEXICO
*:          INSTITUTO TECNOLOGICO DE LA LAGUNA
*:          INGENIERIA EN SISTEMAS COMPUTACIONALES
*:          TOPICOS AVANZADOS DE PROGRAMACION "B"
*:
*:          SEMESTRE: AGO-DIC/2020      HORA: 17-18 HRS
*:
*:          Clase que maneja los metodos para acceder a la información de la base de datos.
*:
*: Archivo      : EjecutorSQL.java
*: Autor        : Félix Gerardo Martínez Hinojo      17130800
*: Fecha        : 01/DIC/2020
*: Compilador   : JAVA J2SE v1.8.0
*: Descripción  : Clase con la funcionalidad para conectarnos y accedernos a la base de datos desde la app.
*:
*: Ultima modif:
*: Fecha        Modificación      Motivo
*:-----
*: 01/DIC/2020 Félix Mtz          Se añadió el prologo, y los metodos para las conexiones a las bases de datos
*: y la obtencion de los datos.
*:-----*/
package mx.tecnm.itl.jdbc;

import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import mx.tecnm.itl.jdbc.ConexionDB;

public class EjecutorSQL {

    public static final String INT      = "int";
    public static final String FLOAT    = "float";
    public static final String STRING   = "string";

    // -----
    public static ResultSet sqlQuery ( String sql ) throws SQLException {
        return sqlQuery ( sql, null );
    }

    // -----
    public static ResultSet sqlQuery ( String sql, Object [][] args ) throws SQLException {
        PreparedStatement ps = ConexionDB.getInstancia().getConexion()
            .prepareStatement ( sql );
        prepararArgumentos ( ps, args );
        return ps.executeQuery ();
    }

    // -----
    public static int sqlEjecutar ( String sql ) throws SQLException {
        return sqlEjecutar ( sql, null );
    }

    // -----
    public static int sqlEjecutar ( String sql, Object [][] args ) throws SQLException {
        PreparedStatement ps = ConexionDB.getInstancia().getConexion()
            .prepareStatement ( sql );

        prepararArgumentos ( ps, args );

        return ps.executeUpdate ();
    }

    // -----

```

```
public static void prepararArgumentos ( PreparedStatement ps, Object[][] args ) throws SQLException {  
    final int TIPO = 0;  
    final int VALOR = 1;  
  
    // Verificar si se debe realizar la sustitución de parametros  
    if ( args != null ) {  
        int numArg = 1;  
        for ( Object[] arg : args ) {  
            switch ( arg [ TIPO ].toString () ) {  
                case INT : ps.setInt ( numArg, Integer.parseInt ( arg [ VALOR ].toString() ) );  
                    break;  
                case FLOAT : ps.setFloat ( numArg, Float.parseFloat ( arg [ VALOR ].toString() ) );  
                    break;  
                case STRING : ps.setString ( numArg, arg [ VALOR ].toString() );  
                    break;  
            }  
            numArg++;  
        }  
    }  
}
```


PrincipalFrame.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Formulario de Reconexión a bases de datos. Vista Principal de la app.
.*
.* Archivo   : PrincipalFrame.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Clase Frame con la parte visual programada de la reconexión a la base de datos, entre la cual
.*              podemos acceder a la base de datos y
.*              Manejar información de categorías como Alumnos, Materias, Kardex, etc.
.*
.*              - ID Materia
.*              - Materia
.*              - Creditos
.*              a través de su clase pertinente.
.* Última modif:
.* Fecha      Modific  Motivo
.*=====
.*
.* 01/DIC/2020 Félix Mtz   Se añadió el dialog al proyecto y se le dió la definición básica..
.* 03/DIC/2020 Félix Mtz   Se añadieron los metodos pertinentes para acceder a la información.
.* 04/DIC/2020 Félix Mtz   Añadida la funcionalidad Del Menú toolbar - Inicio
.* 07/DIC/2020 Félix Mtz   Añadida la funcionalidad Del Sidebar - Mostrar Alumnos
.* 09/DIC/2020 Félix Mtz   Añadida la funcionalidad Del Sidebar - Mostrar Materias
.* 10/DIC/2020 Félix Mtz   Añadida la funcionalidad Del Sidebar - Mostrar Kardex
.* 11/DIC/2020 Félix Mtz   Se añade la funcionalidad de Filtrar - Toolbar Menu
.* 11/DIC/2020 Félix Mtz   Se añade la funcionalidad de Editar - Toolbar Menu
.* 11/DIC/2020 Félix Mtz   Se añade la funcionalidad de Nuevo - Toolbar Menu
.* 14/DIC/2020 Félix Mtz   Se edita y termina la funcionalidad de el Menu Toolbar.
.* 15/DIC/2020 Félix Mtz   Se añadió funcionalidad nueva, como la funcion RecalcularPromedios para actualizar los
.*                          promedios de manera automática.
.*-----*/

package app;

import java.awt.Color;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.SQLIntegrityConstraintViolationException;
import java.util.Properties;
import java.util.Vector;

import javax.swing.JButton;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import modelo.Modelo1;
import modelo.Modelo2;

```

```
import modelo.Modelo3;
import mx.tecnm.itl.extras.AcercaDeDialog;
import mx.tecnm.itl.jdbc.*;
```

```
public class PrincipalFrame extends javax.swing.JFrame {

    public static final String NUEVO = "Nuevo";
    public static final String EDITAR = "Editar";

    public static final String TIT_FRAME          = "Sistema De Calificaciones";
    public static final String TIT_INICIO         = "Porterazos Inc";
    public static final String TIT_MODULO1        = "Alumnos";
    public static final String TIT_MODULO2        = "Materias";
    public static final String TIT_MODULO3        = "Kardex";

    public static final String ALUMNOS_TODOS_POR_NOMBRE      = "alumnos_todos_por_nombre";
    public static final String ALUMNOS_TODOS_SIN_ORDEN      = "alumnos_todos_sin_orden";
    public static final String ALUMNOS_ELIMINAR_X_MATRICULA  = "alumnos_eliminar_x_matricula";
    public static final String ALUMNOS_ACTUALIZA_DATOS       = "alumnos_actualiza_datos";
    public static final String ALUMNOS_INSERTA_NUEVO        = "alumnos_inserta_nuevo";
    public static final String ALUMNOS_ACTUALIZA_PROMEDIOS   = "alumnos_actualiza_promedios";

    public static final String MATERIAS_TODOS_POR_MATERIA    = "materias_todos_por_materia";
    public static final String MATERIAS_TODOS_SIN_ORDEN      = "materias_todos_sin_orden";
    public static final String MATERIAS_ELIMINAR_X_IDMATERIA  = "materias_eliminar_x_IDMateria";

    public static final String MATERIAS_ACTUALIZA_DATOS      = "materias_actualiza_datos";
    public static final String MATERIAS_INSERTA_NUEVO        = "materias_inserta_nuevo";

    public static final String KARDEX_TODOS_POR_MATRICULA    = "kardex_todos_por_matricula";
    public static final String KARDEX_TODOS_SIN_ORDEN        = "kardex_todos_sin_orden";
    public static final String KARDEX_ELIMINAR_X_IDCALIFIC   = "kardex_eliminar_x_IDCalific";

    public static final String KARDEX_ACTUALIZA_DATOS        = "kardex_actualiza_datos";
    public static final String KARDEX_INSERTA_NUEVO          = "kardex_inserta_nuevo";
    public static final String KARDEX_OBT_MAX_ID             = "kardex_obt_max_id";

    private String          moduloActual;
    private Vector<String>  vecNombresColumnas;
    private Vector<String>  vecNombresColumnasBD;
    private Vector<String>  vecTiposColumnas;
    private DefaultTableModel dtmPrincipal;
    private Properties      propConsultasSQL;
    private int             totRegistros;

    // -----
    public PrincipalFrame() {
        initComponents();

        this.setTitle          ( TIT_FRAME );
        lblMensajesSistema.setText ( "" );
        jtoolbarPrincipal.setVisible ( false );
        jpanelTabla.setVisible   ( false );
        jpanelLogoPrincipal.setVisible ( true );
    }
}
```

```

jlblLeyendaPrincipal.setText ( TIT_INICIO );
jlblModulo1.setText ( TIT_MODULO1 );
jlblModulo2.setText ( TIT_MODULO2 );
jlblModulo3.setText ( TIT_MODULO3 );

jmniModulo1.setText ( TIT_MODULO1 );
jmniModulo2.setText ( TIT_MODULO2 );
jmniModulo3.setText ( TIT_MODULO3 );

prepararSentenciasSQL ();

new ConexionDBFrame ( this ).setVisible( true );

}
// -----

private void prepararSentenciasSQL ( ) {
    propConsultasSQL = new Properties ();
    // -----
    // ALUMNOS
    propConsultasSQL.put ( ALUMNOS_TODOS_POR_NOMBRE, "SELECT * FROM Alumnos ORDER BY nombre"
);
    propConsultasSQL.put ( ALUMNOS_TODOS_SIN_ORDEN, "SELECT * FROM Alumnos" );
    propConsultasSQL.put ( ALUMNOS_ELIMINAR_X_MATRICULA, "DELETE FROM Alumnos WHERE
MATRICULA = ?" );

    propConsultasSQL.put ( ALUMNOS_ACTUALIZA_DATOS, "UPDATE Alumnos SET nombre = ? , apellidos = ? ,
edad = ? , promedio = ? " +
        " WHERE matricula = ? " );
    propConsultasSQL.put ( ALUMNOS_INSERTA_NUEVO, "INSERT INTO Alumnos VALUES ( ?, ?, ? , ? , ? )" );

    propConsultasSQL.put ( ALUMNOS_ACTUALIZA_PROMEDIOS, "UPDATE Alumnos a SET PROMEDIO = ( "
        + "SELECT AVG ( CALIFICACION ) FROM Kardex k "
        + "WHERE k.MATRICULA = a.MATRICULA )" );

    // -----
    // MATERIAS
    propConsultasSQL.put ( MATERIAS_TODOS_POR_MATERIA, "SELECT * FROM Materias ORDER BY materia"
);
    propConsultasSQL.put ( MATERIAS_TODOS_SIN_ORDEN, "SELECT * FROM Materias" );
    propConsultasSQL.put ( MATERIAS_ELIMINAR_X_IDMATERIA, "DELETE FROM Materias WHERE IDMATERIA
= ?" );

    propConsultasSQL.put ( MATERIAS_ACTUALIZA_DATOS, "UPDATE Materias SET materia = ? , credits = ? " +
        " WHERE idmateria = ? " );
    propConsultasSQL.put ( MATERIAS_INSERTA_NUEVO, "INSERT INTO Materias VALUES ( ?, ?, ? )" );

    // -----
    // KARDEX
    propConsultasSQL.put ( KARDEX_ELIMINAR_X_IDCALIFIC, "DELETE FROM Kardex WHERE IDCALIFIC = ?"
);
    propConsultasSQL.put ( KARDEX_TODOS_SIN_ORDEN,
        "SELECT k.IDCALIFIC, a.MATRICULA, a.NOMBRE, a.APELLIDOS, m.IDMATERIA, m.MATERIA,
k.PERIODO, k.CALIFICACION " +
        "FROM Alumnos a JOIN KARDEX k ON a.MATRICULA = k.MATRICULA " +
        "JOIN Materias m ON m.IDMATERIA = k.IDMATERIA"

```

```

    );
    propConsultasSQL.put ( KARDEX_TODOS_POR_MATRICULA,
        "SELECT k.IDCALIFIC, a.MATRICULA, a.NOMBRE, a.APELLIDOS,m.IDMATERIA, m.MATERIA ,
        k.PERIODO, k.CALIFICACION \n" +
        "    FROM Alumnos a, Kardex k , Materias m WHERE a.MATRICULA = k.MATRICULA AND k.IDMATERIA =
        m.IDMATERIA \n" +
        "    ORDER BY k.MATRICULA" );

    propConsultasSQL.put ( KARDEX_ACTUALIZA_DATOS, "UPDATE Kardex SET MATRICULA = ? , IDMATERIA
    = ?, PERIODO = ?, CALIFICACION = ? "
        + "WHERE IDCALIFIC = ?" );
    propConsultasSQL.put ( KARDEX_INSERTA_NUEVO, "INSERT INTO Kardex VALUES ( ?, ?, ?, ?, ? )" );
    propConsultasSQL.put ( KARDEX_OBT_MAX_ID, "SELECT MAX ( IDCALIFIC ) AS maxid FROM Kardex" );
}

```

```
@SuppressWarnings("unchecked")
```

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

```
private void initComponents() {
```

```

    jtoolbarPrincipal = new javax.swing.JToolBar();
    btnNuevo = new javax.swing.JButton();
    btnEditar = new javax.swing.JButton();
    btnEliminar = new javax.swing.JButton();
    btnInicio = new javax.swing.JButton();
    btnFiltrar = new javax.swing.JButton();
    jPanelBarraEstado = new javax.swing.JPanel();
    jtoolbarEstatus = new javax.swing.JToolBar();
    jLabel1 = new javax.swing.JLabel();
    lblEstatusConexion = new javax.swing.JLabel();
    jtoolbarMensajes = new javax.swing.JToolBar();
    lblMensajesSistema = new javax.swing.JLabel();
    jscrollpaneBarraModulosBotones = new javax.swing.JScrollPane();
    jPanelBarraModulos = new javax.swing.JPanel();
    btnModulo1 = new javax.swing.JButton();
    btnModulo2 = new javax.swing.JButton();
    btnModulo3 = new javax.swing.JButton();
    lblModulo1 = new javax.swing.JLabel();
    lblModulo2 = new javax.swing.JLabel();
    lblModulo3 = new javax.swing.JLabel();
    jPanelLeyendaPrincipal = new javax.swing.JPanel();
    lblLeyendaPrincipal = new javax.swing.JLabel();
    jPanelTabla = new javax.swing.JPanel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTablePrincipal = new javax.swing.JTable();
    jPanelLogoPrincipal = new javax.swing.JPanel();
    lblLogoSistema = new javax.swing.JLabel();
    jMenuBar1 = new javax.swing.JMenuBar();
    jMenu1 = new javax.swing.JMenu();
    jmi_Salir = new javax.swing.JMenuItem();
    jMenu3 = new javax.swing.JMenu();
    jmnReportes = new javax.swing.JMenu();
    jmnModulo1 = new javax.swing.JMenuItem();
    jmnModulo2 = new javax.swing.JMenuItem();
    jmnModulo3 = new javax.swing.JMenuItem();
    jmnRecalcularPromedios = new javax.swing.JMenuItem();
    jMenu2 = new javax.swing.JMenu();
    jmi_AcercaDe = new javax.swing.JMenuItem();

```

```
setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setExtendedState(6);
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowActivated(java.awt.event.WindowEvent evt) {
        formWindowActivated(evt);
    }
});

jtoolbarPrincipal.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jtoolbarPrincipal.setRollover(true);

jbtnNuevo.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/newFile_24px.png"))); // NOI18N
jbtnNuevo.setText("Nuevo");
jbtnNuevo.setFocusable(false);
jbtnNuevo.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
jbtnNuevo.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
jbtnNuevo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnNuevoActionPerformed(evt);
    }
});
jtoolbarPrincipal.add(jbtnNuevo);

jbtnEditar.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/edit_24px.png"))); // NOI18N
jbtnEditar.setText("Editar");
jbtnEditar.setFocusable(false);
jbtnEditar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
jbtnEditar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
jbtnEditar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnEditarActionPerformed(evt);
    }
});
jtoolbarPrincipal.add(jbtnEditar);

jbtnEliminar.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/delete_24x.png"))); // NOI18N
jbtnEliminar.setText("Eliminar");
jbtnEliminar.setFocusable(false);
jbtnEliminar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
jbtnEliminar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
jbtnEliminar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnEliminarActionPerformed(evt);
    }
});
jtoolbarPrincipal.add(jbtnEliminar);

jbtnInicio.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/home_24px.png"))); // NOI18N
jbtnInicio.setText("Inicio");
jbtnInicio.setFocusable(false);
jbtnInicio.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
jbtnInicio.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
jbtnInicio.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnInicioActionPerformed(evt);
    }
});
jtoolbarPrincipal.add(jbtnInicio);
```

```
jbtnFiltrar.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/filter.png"))); // NOI18N
jbtnFiltrar.setText("Filtrar");
jbtnFiltrar.setFocusable(false);
jbtnFiltrar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
jbtnFiltrar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
jbtnFiltrar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnFiltrarActionPerformed(evt);
    }
});
jtoolbarPrincipal.add(jbtnFiltrar);

jtoolbarEstatus.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jtoolbarEstatus.setRollover(true);

jLabel1.setText("Estatus de la conexión.");
jtoolbarEstatus.add(jLabel1);

jlblEstatusConexion.setText("<Estatus>");
jtoolbarEstatus.add(jlblEstatusConexion);

jtoolbarMensajes.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jtoolbarMensajes.setRollover(true);

jlblMensajesSistema.setText("<Mensajes>");
jtoolbarMensajes.add(jlblMensajesSistema);

javax.swing.GroupLayout jPanelBarraEstadoLayout = new javax.swing.GroupLayout(jPanelBarraEstado);
jpanelBarraEstado.setLayout(jPanelBarraEstadoLayout);
jpanelBarraEstadoLayout.setHorizontalGroup(
    jPanelBarraEstadoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanelBarraEstadoLayout.createSequentialGroup()
            .addComponent(jtoolbarEstatus, javax.swing.GroupLayout.PREFERRED_SIZE, 400,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jtoolbarMensajes, javax.swing.GroupLayout.PREFERRED_SIZE, 483,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap())
        );
jpanelBarraEstadoLayout.setVerticalGroup(
    jPanelBarraEstadoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanelBarraEstadoLayout.createSequentialGroup()
            .addGroup(jPanelBarraEstadoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .add(jtoolbarEstatus, javax.swing.GroupLayout.PREFERRED_SIZE, 400,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .add(jtoolbarMensajes, javax.swing.GroupLayout.PREFERRED_SIZE, 483,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap())
        );

jbtnModulo1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/Student_32px.png"))); //
NOI18N
jbtnModulo1.setToolTipText("Alumnos");
jbtnModulo1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        jbtnModulo1ActionPerformed(evt);
    }
});

jbtnModulo2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/book_32px.png"))); // NOI18N
jbtnModulo2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnModulo2ActionPerformed(evt);
    }
});

jbtnModulo3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/Kardex_32px.png"))); //
NOI18N
jbtnModulo3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnModulo3ActionPerformed(evt);
    }
});

jlblModulo1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jlblModulo1.setText("Modulo 1");

jlblModulo2.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jlblModulo2.setText("Modulo 2");

jlblModulo3.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
jlblModulo3.setText("Modulo 3");

javax.swing.GroupLayout jPanelBarraModulosLayout = new javax.swing.GroupLayout(jPanelBarraModulos);
jpanelBarraModulos.setLayout(jPanelBarraModulosLayout);
jpanelBarraModulosLayout.setHorizontalGroup(
    jPanelBarraModulosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanelBarraModulosLayout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(jPanelBarraModulosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jbtnModulo1, javax.swing.GroupLayout.DEFAULT_SIZE, 79, Short.MAX_VALUE)
                .addComponent(jbtnModulo2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jbtnModulo3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            )
            .addGap(10, 10, 10)
        )
        .addGroup(jPanelBarraModulosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanelBarraModulosLayout.createSequentialGroup()
                .addComponent(jlblModulo1)
                .addComponent(jlblModulo2)
                .addComponent(jlblModulo3)
                .addGap(0, 0, Short.MAX_VALUE))
            .addGap(10, 10, 10)
        )
);
jpanelBarraModulosLayout.setVerticalGroup(
    jPanelBarraModulosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanelBarraModulosLayout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addComponent(jbtnModulo1, javax.swing.GroupLayout.PREFERRED_SIZE, 53,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jlblModulo1)
            .addGap(36, 36, 36)
        )
);
```

```
.addComponent(jbtnModulo2, javax.swing.GroupLayout.PREFERRED_SIZE, 53,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addGap(18, 18, 18)
.addComponent(jlblModulo2)
.addGap(24, 24, 24)
.addComponent(jbtnModulo3, javax.swing.GroupLayout.PREFERRED_SIZE, 53,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addGap(18, 18, 18)
.addComponent(jlblModulo3)
.addContainerGap(308, Short.MAX_VALUE))
);

jscrollpaneBarraModulosBotones.setViewportView(jpanelBarraModulos);

jpanelLeyendaPrincipal.setBackground(new java.awt.Color(0, 0, 153));

jlblLeyendaPrincipal.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jlblLeyendaPrincipal.setForeground(new java.awt.Color(255, 255, 255));
jlblLeyendaPrincipal.setText("Modulo");

javax.swing.GroupLayout jpanelLeyendaPrincipalLayout = new
javax.swing.GroupLayout(jpanelLeyendaPrincipal);
jpanelLeyendaPrincipal.setLayout(jpanelLeyendaPrincipalLayout);
jpanelLeyendaPrincipalLayout.setHorizontalGroup(
    jpanelLeyendaPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jpanelLeyendaPrincipalLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jlblLeyendaPrincipal)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
);
jpanelLeyendaPrincipalLayout.setVerticalGroup(
    jpanelLeyendaPrincipalLayout.createSequentialGroup()
        .addGroup(jpanelLeyendaPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .add(jpanelLeyendaPrincipalLayout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jlblLeyendaPrincipal)
                .addGap(19, 19, 19))
        );

jpanelTabla.setBorder(javax.swing.BorderFactory.createEtchedBorder());

jtablePrincipal.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
jScrollPane1.setViewportView(jtablePrincipal);

javax.swing.GroupLayout jpanelTablaLayout = new javax.swing.GroupLayout(jpanelTabla);
jpanelTabla.setLayout(jpanelTablaLayout);
jpanelTablaLayout.setHorizontalGroup(
    jpanelTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```



```
.addComponent(jScrollPane1, javax.swing.GroupLayout.Alignment.TRAILING)
);
jpanelTablaLayout.setVerticalGroup(
    jpanelTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 191, Short.MAX_VALUE)
);

jlblLogoSistema.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jlblLogoSistema.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/school_big.png"))); //
NOI18N

javax.swing.GroupLayout jPanelLogoPrincipalLayout = new javax.swing.GroupLayout(jPanelLogoPrincipal);
jpanelLogoPrincipal.setLayout(jPanelLogoPrincipalLayout);
jpanelLogoPrincipalLayout.setHorizontalGroup(
    jPanelLogoPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanelLogoPrincipalLayout.createSequentialGroup())
    .addContainerGap()
    .addComponent(jlblLogoSistema, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addContainerGap()
);
jpanelLogoPrincipalLayout.setVerticalGroup(
    jPanelLogoPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanelLogoPrincipalLayout.createSequentialGroup())
    .addContainerGap()
    .addComponent(jlblLogoSistema, javax.swing.GroupLayout.DEFAULT_SIZE, 264, Short.MAX_VALUE)
    .addContainerGap()
);

jMenu1.setText("Archivo");

jmi_Salir.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_F4,
java.awt.event.InputEvent.ALT_MASK));
jmi_Salir.setText("Salir");
jmi_Salir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jmi_SalirActionPerformed(evt);
    }
});
jMenu1.add(jmi_Salir);

jMenuBar1.add(jMenu1);

jMenu3.setText("Herramientas");

jmnReportes.setText("Reportes");

jmniModulo1.setText("Modulo1");
jmnReportes.add(jmniModulo1);

jmniModulo2.setText("Modulo2");
jmnReportes.add(jmniModulo2);

jmniModulo3.setText("Modulo3");
jmnReportes.add(jmniModulo3);

jMenu3.add(jmnReportes);
```

```
jmnRecalcularPromedios.setText("Recalcular Promedios");
jmnRecalcularPromedios.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jmnRecalcularPromediosActionPerformed(evt);
    }
});
jMenu3.add(jmnRecalcularPromedios);

jMenuBar1.add(jMenu3);

jMenu2.setText("Ayuda");

jmi_AcercaDe.setText("Acerca De");
jmi_AcercaDe.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jmi_AcercaDeActionPerformed(evt);
    }
});
jMenu2.add(jmi_AcercaDe);

jMenuBar1.add(jMenu2);

setJMenuBar(jMenuBar1);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jtoolbarPrincipal, javax.swing.GroupLayout.DEFAULT_SIZE, 951, Short.MAX_VALUE)
        .addComponent(jpanelBarraEstado, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jscrollpaneBarraModulosBotones, javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jpanelLeyendaPrincipal, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jpanelTabla, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jpanelLogoPrincipal, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jtoolbarPrincipal, javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jpanelLeyendaPrincipal, javax.swing.GroupLayout.PREFERRED_SIZE, 46,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(11, 11, 11)
                    .addComponent(jpanelTabla, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(jpanelLogoPrincipal, javax.swing.GroupLayout.PREFERRED_SIZE,
```

```

javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jscrollpaneBarraModulosBotones, javax.swing.GroupLayout.PREFERRED_SIZE, 0,
Short.MAX_VALUE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jpanelBarraEstado, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    );

    pack();
} // </editor-fold>
// -----
private void formWindowActivated(java.awt.event.WindowEvent evt) {
    if ( ConexionDB.getInstancia().conectado () ) {
        lblEstatusConexion.setText ( "Conectado" );
        lblEstatusConexion.setForeground ( Color.BLUE );
    } else {
        lblEstatusConexion.setForeground( Color.RED );
    }
}
// -----
private void jmi_SalirActionPerformed(java.awt.event.ActionEvent evt) {
    ConexionDB.getInstancia().desconectar ();
    dispose ();
}
// -----
private void jbtnModulo1ActionPerformed(java.awt.event.ActionEvent evt) {
    prepararVistaModulo ( TIT_MODULO1 );

    String sql = propConsultasSQL.getProperty ( ALUMNOS_TODOS_POR_NOMBRE );
    desplegarRegistros ( sql, null );
}
// -----
private void jbtnModulo2ActionPerformed(java.awt.event.ActionEvent evt) {
    prepararVistaModulo ( TIT_MODULO2 );

    String sql = propConsultasSQL.getProperty ( MATERIAS_TODOS_POR_MATERIA );
    desplegarRegistros ( sql, null );
}
// -----
private void jbtnModulo3ActionPerformed(java.awt.event.ActionEvent evt) {
    prepararVistaModulo ( TIT_MODULO3 );

    String sql = propConsultasSQL.getProperty ( KARDEX_TODOS_POR_MATRICULA );
    desplegarRegistros ( sql, null );
}
// -----
private void jbtnInicioActionPerformed(java.awt.event.ActionEvent evt) {
    jtoolbarPrincipal.setVisible      ( false );
    jpanelLogoPrincipal.setVisible    ( true );
    jpanelTabla.setVisible            ( false );

    lblLeyendaPrincipal.setText      ( TIT_INICIO );
    moduloActual                      = "";

    lblMensajesSistema.setText ( "" );
}
// -----

```

```
private void jbtnFiltrarActionPerformed(java.awt.event.ActionEvent evt) {
    FiltrarDialog fd = new FiltrarDialog ( this, true );
    fd.setVisible ( true );
}
// -----
private void jbtnEliminarActionPerformed(java.awt.event.ActionEvent evt) {
    int pos = jTablePrincipal.getSelectedRow ();
    if ( pos == -1 ) {
        JOptionPane.showMessageDialog ( this,
            "No hay un registro seleccionado",
            "Eliminar",
            JOptionPane.ERROR_MESSAGE
        );
        return;
    }

    int confirma = JOptionPane.showConfirmDialog (
        this,
        "Eliminar el registro seleccionado? ",
        "Eliminar",
        JOptionPane.YES_NO_OPTION,
        JOptionPane.QUESTION_MESSAGE
    );

    if ( confirma == JOptionPane.NO_OPTION )
        return;

    String valorLlavePrim = jTablePrincipal.getValueAt ( pos, 0 ).toString ();
    String tipoLlavePrim = vecTiposColumnas.elementAt ( 0 );
    String sql = "";
    JButton jbtnModulo = null;

    if ( moduloActual == TIT_MODULO1 ) {
        sql = propConsultasSQL.getProperty ( ALUMNOS_ELIMINAR_X_MATRICULA );
        jbtnModulo = jbtnModulo1;
    } else if ( moduloActual == TIT_MODULO2 ) {
        sql = propConsultasSQL.getProperty ( MATERIAS_ELIMINAR_X_IDMATERIA );
        jbtnModulo = jbtnModulo2;
    } else if ( moduloActual == TIT_MODULO3 ) {
        sql = propConsultasSQL.getProperty ( KARDEX_ELIMINAR_X_IDCALIFIC );
        jbtnModulo = jbtnModulo3;
    }

    Object[][] args = { { tipoLlavePrim, valorLlavePrim } };

    try {
        int regs = EjecutorSQL.sqlEjecutar ( sql, args );
        if ( regs == 1 ) {
            jbtnModulo.doClick ();
            JOptionPane.showMessageDialog ( this, "Se eliminó el registro correctamente", "Error",
                JOptionPane.INFORMATION_MESSAGE );
        }
    } catch ( SQLIntegrityConstraintViolationException ex ) {
        JOptionPane.showMessageDialog (
            this,
            "No se puede eliminar el registro actual porque tiene registros asociados en otras tablas",
            "Error",
            JOptionPane.ERROR_MESSAGE );
    }
}
```

```
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog ( this, ex, "Error", JOptionPane.ERROR_MESSAGE );
    }
}
// -----
private void jbtnEditarActionPerformed(java.awt.event.ActionEvent evt) {
    int pos = jTablePrincipal.getSelectedRow ();
    if ( pos == -1 ) {
        JOptionPane.showMessageDialog ( this,
            "No hay un registro seleccionado",
            "Eliminar",
            JOptionPane.ERROR_MESSAGE
        );
        return;
    }

    if ( moduloActual.equals ( TIT_MODULO1 ) ) {
        String matricula      = jTablePrincipal.getValueAt ( pos, 0 ).toString ();
        String nombre         = jTablePrincipal.getValueAt ( pos, 1 ).toString ();
        String apellidos      = jTablePrincipal.getValueAt ( pos, 2 ).toString ();
        int edad              = Integer.parseInt ( jTablePrincipal.getValueAt ( pos, 3 ).toString () );
        float promedio        = Float.parseFloat ( jTablePrincipal.getValueAt ( pos, 4 ).toString () );

        Modelo1 modelo1 = new Modelo1 ( matricula, nombre, apellidos, edad, promedio );

        Modulo1EditarDialog dialog = new Modulo1EditarDialog ( this, modelo1 );
        dialog.setVisible ( true );

    } else if ( moduloActual.equals ( TIT_MODULO2 ) ) {
        String idmateria      = jTablePrincipal.getValueAt ( pos, 0 ).toString ();
        String materia        = jTablePrincipal.getValueAt ( pos, 1 ).toString ();
        int creditos          = Integer.parseInt ( jTablePrincipal.getValueAt ( pos, 2 ).toString () );

        Modelo2 modelo2 = new Modelo2 ( idmateria, materia, creditos );

        Modulo2EditarDialog dialog = new Modulo2EditarDialog ( this, modelo2 );
        dialog.setVisible ( true );

    } else if ( moduloActual.equals ( TIT_MODULO3 ) ) {
        int idcalific         = Integer.parseInt (
            jTablePrincipal.getValueAt ( pos, 0 ).toString () );

        String matricula      = jTablePrincipal.getValueAt ( pos, 1 ).toString ();
        String idmateria       = jTablePrincipal.getValueAt ( pos, 4 ).toString ();
        String periodo        = jTablePrincipal.getValueAt ( pos, 6 ).toString ();
        String califica        = jTablePrincipal.getValueAt ( pos, 7 ).toString ();
        int calificacion       = Integer.parseInt ( califica );

        Modelo3 modelo = new Modelo3 ( idcalific, matricula, idmateria, periodo, calificacion );

        Modulo3EditarDialog dialog = new Modulo3EditarDialog ( this, modelo );
        dialog.setVisible ( true );
    }
}
// -----
private void jbtnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
    if ( moduloActual.equals ( TIT_MODULO1 ) ) {
        Modulo1EditarDialog dialog = new Modulo1EditarDialog ( this, null );
        dialog.setVisible ( true );
    }
}
```

```

    } else if ( moduloActual.equals ( TIT_MODULO2 ) ) {
        Modulo2EditarDialog dialog = new Modulo2EditarDialog ( this, null );
        dialog.setVisible ( true );
    }
    else if ( moduloActual.equals ( TIT_MODULO3 ) ) {
        Modulo3EditarDialog dialog = new Modulo3EditarDialog ( this, null );
        dialog.setVisible ( true );
    }
}
// -----
private void jmnRecalcularPromediosActionPerformed(java.awt.event.ActionEvent evt) {
    String sql = propConsultasSQL.getProperty ( PrincipalFrame.ALUMNOS_ACTUALIZA_PROMEDIOS );
    try {
        int regs = EjecutorSQL.sqlEjecutar ( sql );

        JOptionPane.showMessageDialog (
            this,
            "Proceso Terminado " + regs + " Promedios Fueron Actualizados",
            "Promedios",
            JOptionPane.INFORMATION_MESSAGE
        );

        this.jbtnModulo1.doClick ();
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog ( this, ex, "Promedios", JOptionPane.ERROR_MESSAGE );
    }
}
// -----
private void jmi_AcercaDeActionPerformed(java.awt.event.ActionEvent evt) {
    jAcercaDeDialog jad_dialog = new jAcercaDeDialog ( this, true );
    jad_dialog.setVisible ( true );
}
// -----

private void prepararVistaModulo ( String modulo ) {
    moduloActual = modulo;
    jtoolbarPrincipal.setVisible      ( true );
    jpanelLogoPrincipal.setVisible    ( false );
    jpanelTabla.setVisible            ( true );

    lblLeyendaPrincipal.setText      ( modulo );

    determinarNombresColumnas        ( modulo );

    dtmPrincipal = new DefaultTableModel ( vecNombresColumnas, 0 );
    jTablePrincipal.setModel           ( dtmPrincipal );
}
// -----
private void determinarNombresColumnas ( String modulo ) {
    vecNombresColumnas = new Vector<String> ();
    vecNombresColumnasBD = new Vector<String> ();
    vecTiposColumnas = new Vector<String> ();
    if ( modulo.equals ( TIT_MODULO1 ) ) {
        vecNombresColumnas.add ( "Matricula" );
        vecNombresColumnas.add ( "Nombre" );
        vecNombresColumnas.add ( "Apellidos" );
        vecNombresColumnas.add ( "Edad" );
        vecNombresColumnas.add ( "Promedio" );
    }
}

```

```

vecNombresColumnasBD.add ( "Matricula" );
vecNombresColumnasBD.add ( "Nombre" );
vecNombresColumnasBD.add ( "Apellidos" );
vecNombresColumnasBD.add ( "Edad" );
vecNombresColumnasBD.add ( "Promedio" );

vecTiposColumnas.add( EjecutorSQL.STRING );
vecTiposColumnas.add( EjecutorSQL.STRING );
vecTiposColumnas.add( EjecutorSQL.STRING );
vecTiposColumnas.add( EjecutorSQL.INT );
vecTiposColumnas.add( EjecutorSQL.FLOAT );

} else if ( modulo.equals ( TIT_MODULO2 ) ) {
    vecNombresColumnas.add ( "ID Materia" );
    vecNombresColumnas.add ( "Materia" );
    vecNombresColumnas.add ( "Creditos" );

    vecNombresColumnasBD.add ( "IDMateria" );
    vecNombresColumnasBD.add ( "Materia" );
    vecNombresColumnasBD.add ( "Creditos" );

    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.INT );

} else if ( modulo.equals ( TIT_MODULO3 ) ) {
    vecNombresColumnas.add ( "ID Calific" );
    vecNombresColumnas.add ( "Matricula" );
    vecNombresColumnas.add ( "Nombre" );
    vecNombresColumnas.add ( "Apellidos" );
    vecNombresColumnas.add ( "ID Materia" );
    vecNombresColumnas.add ( "Materia" );
    vecNombresColumnas.add ( "Periodo" );
    vecNombresColumnas.add ( "Calificación" );

    vecNombresColumnasBD.add ( "k.IDCALIFIC" );
    vecNombresColumnasBD.add ( "a.MATRICULA" );
    vecNombresColumnasBD.add ( "a.NOMBRE" );
    vecNombresColumnasBD.add ( "a.APELLIDOS" );
    vecNombresColumnasBD.add ( "m.IDMATERIA" );
    vecNombresColumnasBD.add ( "m.MATERIA" );
    vecNombresColumnasBD.add ( "k.PERIODO" );
    vecNombresColumnasBD.add ( "k.CALIFICACION" );

    vecTiposColumnas.add( EjecutorSQL.INT );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.STRING );
    vecTiposColumnas.add( EjecutorSQL.INT );
}
}
// -----
protected void desplegarRegistros ( String sql, Object[][] args ) {

```

```

ResultSet rs;
try {

    rs = EjecutorSQL.sqlQuery ( sql, args );

    dtmPrincipal = new DefaultTableModel ( vecNombresColumnas, 0 );
    while ( rs.next () ) {
        Object[] fila = crearFila( rs );
        dtmPrincipal.addRow( fila );

    }
    rs.close ();
    jTablePrincipal.setModel ( dtmPrincipal );
    totRegistros = dtmPrincipal.getRowCount ();
    lblMensajesSistema.setText ( totRegistros + " registros" );
} catch (SQLException e) {
    JOptionPane.showMessageDialog ( this, e, " Error ", JOptionPane.ERROR_MESSAGE );
}
}
// -----
private Object [] crearFila ( ResultSet rs ) throws SQLException {
    if ( moduloActual.equals ( TIT_MODULO1 ) ) {
        String matricula = rs.getString ( "matricula" );
        String nombre = rs.getString ( "nombre" );
        String apellidos = rs.getString ( "apellidos" );
        int edad = rs.getInt ( "edad" );
        float promedio = rs.getFloat ( "promedio" );

        Object [] fila = { matricula, nombre, apellidos, edad, promedio };
        return fila;
    }
    else if ( moduloActual.equals ( TIT_MODULO2 ) ) {
        String matricula = rs.getString ( "IDMATERIA" );
        String nombre = rs.getString ( "MATERIA" );
        int creditos = rs.getInt ( "CREDITOS" );

        Object [] fila = { matricula, nombre, creditos };
        return fila;
    }
    else if ( moduloActual.equals ( TIT_MODULO3 ) ) {
        int idCalif = rs.getInt ( "IDCALIFIC" );
        String matricula = rs.getString ( "MATRICULA" );
        String nombre = rs.getString ( "NOMBRE" );
        String apellidos = rs.getString ( "APELLIDOS" );
        String idmateria = rs.getString ( "IDMATERIA" );
        String materia = rs.getString ( "MATERIA" );
        String periodo = rs.getString ( "PERIODO" );
        int calificacion = rs.getInt ( "CALIFICACION" );

        Object [] fila = { idCalif, matricula, nombre, apellidos, idmateria, materia, periodo, calificacion };
        return fila;
    }
    return null;
}
// -----

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 ("Windows".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(PrincipalFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(PrincipalFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(PrincipalFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
    java.util.logging.Logger.getLogger(PrincipalFrame.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 PrincipalFrame().setVisible ( false );
    }
});
}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JMenu jMenu1;
private javax.swing.JMenu jMenu2;
private javax.swing.JMenu jMenu3;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JButton btnEditar;
private javax.swing.JButton btnEliminar;
private javax.swing.JButton btnFiltrar;
private javax.swing.JButton btnInicio;
private javax.swing.JButton btnModulo1;
private javax.swing.JButton btnModulo2;
private javax.swing.JButton btnModulo3;
private javax.swing.JButton btnNuevo;
private javax.swing.JLabel lblEstatusConexion;
private javax.swing.JLabel lblLeyendaPrincipal;
private javax.swing.JLabel lblLogoSistema;
private javax.swing.JLabel lblMensajesSistema;
private javax.swing.JLabel lblModulo1;
private javax.swing.JLabel lblModulo2;
private javax.swing.JLabel lblModulo3;
private javax.swing.JMenu jmi_AcercaDe;
private javax.swing.JMenu jmi_Salir;

```

```
private javax.swing.JMenuItem jmnRecalcularPromedios;
private javax.swing.JMenu jmnReportes;
private javax.swing.JMenuItem jmniModulo1;
private javax.swing.JMenuItem jmniModulo2;
private javax.swing.JMenuItem jmniModulo3;
private javax.swing.JPanel jpanelBarraEstado;
private javax.swing.JPanel jpanelBarraModulos;
private javax.swing.JPanel jpanelLeyendaPrincipal;
private javax.swing.JPanel jpanelLogoPrincipal;
private javax.swing.JPanel jpanelTabla;
private javax.swing.JScrollPane jscrollpaneBarraModulosBotones;
private javax.swing.JTable jtablePrincipal;
private javax.swing.JToolBar jtoolbarEstatus;
private javax.swing.JToolBar jtoolbarMensajes;
private javax.swing.JToolBar jtoolbarPrincipal;
// End of variables declaration
public String getModuloActual() {
    return moduloActual;
}

public void setModuloActual( String moduloActual ) {
    this.moduloActual = moduloActual;
}

public Vector<String> getVecNombresColumnas() {
    return vecNombresColumnas;
}

public void setVecNombresColumnas( Vector<String> vecNombresColumnas ) {
    this.vecNombresColumnas = vecNombresColumnas;
}

public Vector<String> getVecNombresColumnasBD() {
    return vecNombresColumnasBD;
}

public void setVecNombresColumnasBD ( Vector<String> vecNombresColumnasBD ) {
    this.vecNombresColumnasBD = vecNombresColumnasBD;
}

public Vector<String> getVecTiposColumnas() {
    return vecTiposColumnas;
}

public void setVecTiposColumnas ( Vector<String> vecTiposColumnas ) {
    this.vecTiposColumnas = vecTiposColumnas;
}

public Properties getPropConsultasSQL() {
    return propConsultasSQL;
}

public void setPropConsultasSQL ( Properties propConsultasSQL ) {
    this.propConsultasSQL = propConsultasSQL;
}

public JButton getJbtnModulo1 () {
    return jbtnModulo1;
}
```

```

    }

    public JButton getJbtnModulo2 () {
        return jbtnModulo2;
    }

    public JButton getJbtnModulo3 () {
        return jbtnModulo3;
    }

}

```

FiltrarDialog.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Dialogo que permite filtrar datos de diferentes categorías.
.*
.* Archivo   : FiltrarDialog.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Dialogo que permite filtrar datos de Alumnos, Materias y Kardex de Alumnos
.*           por:
.*             - Matricula
.*             - Nombre
.*             - Apellidos
.*             - Promedio
.*             - entre otras.
.* Ultima modif:
.* Fecha   Modific  Motivo
.*=====
.* 07/DIC/2020 Félix Mtz   Se añadió el prologo y se le dió la funcionalidad básica de la app.
.* 07/DIC/2020 Félix Mtz   Se añadió la funcionalidad de todo el dialog.
.*-----*/
package app;

import java.util.Vector;
import javax.swing.JOptionPane;
import mx.tecnm.itl.jdbc.EjecutorSQL;

public class FiltrarDialog extends javax.swing.JDialog {

    private PrincipalFrame      frmPrincipal;
    private Vector<String>      vecNombresColumnas;
    private Vector<String>      vecNombresColumnasBD;
    private Vector<String>      vecTiposColumnas;
    private boolean             inicializando = true;

    // -----
    public FiltrarDialog(java.awt.Frame parent, boolean modal) {

```

```

super(parent, modal);
 initComponents();

frmPrincipal = (PrincipalFrame) parent;
vecNombresColumnas = frmPrincipal.getVecNombresColumnas ();
vecNombresColumnasBD = frmPrincipal.getVecNombresColumnasBD ();
vecTiposColumnas = frmPrincipal.getVecTiposColumnas ();

jcboColumna.removeAllItems ();

for ( int i = 0; i < vecNombresColumnas.size (); i++ ) {
    jcboColumna.addItem ( vecNombresColumnas.elementAt ( i ) );
}

cargarComparadores ( 0 );
jcboColumna.setSelectedIndex ( 0 );
jcboColumna.requestFocus ();
inicializando = false;
}
// -----
private void cargarComparadores ( int pos ) {
    jcboComparador.removeAllItems ();

    if ( vecTiposColumnas.elementAt ( pos ).equals ( EjecutorSQL.STRING ) ) {
        jcboComparador.addItem ( "Sea igual a" );
        jcboComparador.addItem ( "Empiece con" );
        jcboComparador.addItem ( "Termine con" );
        jcboComparador.addItem ( "Contenga" );
        jcboComparador.addItem ( "Sea diferente" );
    } else if ( vecTiposColumnas.elementAt ( pos ).equals ( EjecutorSQL.INT ) || vecTiposColumnas.elementAt ( pos
).equals ( EjecutorSQL.FLOAT ) ) {
        jcboComparador.addItem ( "=" );
        jcboComparador.addItem ( "<>" );
        jcboComparador.addItem ( ">" );
        jcboComparador.addItem ( "<" );
        jcboComparador.addItem ( ">=" );
        jcboComparador.addItem ( "<=" );
    }
    jcboComparador.setSelectedIndex ( 0 );
}
// -----

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jcboColumna = new javax.swing.JComboBox<>();
    jcboComparador = new javax.swing.JComboBox<>();
    jtxfValor = new javax.swing.JTextField();
    jbtnAceptar = new javax.swing.JButton();
    jbtnSalir = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
    setTitle("Filtrar Datos");

```

```

jLabel1.setText("Columna");

jLabel2.setText("Comparador");

jLabel3.setText("Valor");

jcboColumna.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Item 1", "Item 2", "Item 3",
"Item 4" }));
jcboColumna.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(java.awt.event.ItemEvent evt) {
        jcboColumna.itemStateChanged(evt);
    }
});

jcboComparador.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Item 1", "Item 2", "Item 3",
"Item 4" }));

jbtnAceptar.setText("Aceptar");
jbtnAceptar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnAceptar.actionPerformed(evt);
    }
});

jbtnSalir.setText("Salir");
jbtnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnSalir.actionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(46, 46, 46)
            .addComponent(jLabel1)
            .addGap(192, 192, 192)
            .addComponent(jLabel2)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jLabel3)
            .addGap(188, 188, 188))
        .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(24, 24, 24)
                    .addComponent(jcboColumna, javax.swing.GroupLayout.PREFERRED_SIZE, 165,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(61, 61, 61)
                    .addComponent(jcboComparador, javax.swing.GroupLayout.PREFERRED_SIZE, 178,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(119, 119, 119)
                    .addComponent(jtxtValor, javax.swing.GroupLayout.PREFERRED_SIZE, 153,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(119, 119, 119))
                .addGroup(layout.createSequentialGroup()
                    .addGap(46, 46, 46)
                    .addComponent(jbtnAceptar)
                    .addGap(192, 192, 192)
                    .addComponent(jbtnSalir)
                    .addGap(188, 188, 188))
            )
            .addContainerGap(188, true))
);

```

```

        .addGap(207, 207, 207)
        .addComponent(jbtnAceptar)
        .addGap(209, 209, 209)
        .addComponent(jbtnSalir)))
        .addContainerGap(105, Short.MAX_VALUE))
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(49, 49, 49)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(jLabel3))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jcboColumna, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jcboComparador, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jtxfValor, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 75, Short.MAX_VALUE)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jbtnAceptar)
                    .addComponent(jbtnSalir))
                .addGap(46, 46, 46))
        );

    pack();
    setLocationRelativeTo(null);
} // </editor-fold>
// -----
private void jbtnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    dispose ();
}
// -----
private void jcboColumnaltemStateChanged(java.awt.event.ItemEvent evt) {
    if ( ! inicializando )
        cargarComparadores ( jcboColumna.getSelectedIndex () );
}
// -----
private void jbtnAceptarActionPerformed(java.awt.event.ActionEvent evt) {

    if ( jtxfValor.getText().trim().equals ( "" ) ) {
        JOptionPane.showMessageDialog(
            this,
            "Proporcione un valor",
            "Filtrar",
            JOptionPane.ERROR_MESSAGE
        );
        jtxfValor.requestFocus ();
        return;
    }

    String sql = "";
    String orderBy = "";

```

```

if ( frmPrincipal.getModuloActual().equals( PrincipalFrame.TIT_MODULO1 ) ) {
    sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.ALUMNOS_TODOS_SIN_ORDEN );
    orderBy = "order by nombre";
} else if ( frmPrincipal.getModuloActual().equals( PrincipalFrame.TIT_MODULO2 ) ) {
    sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.MATERIAS_TODOS_SIN_ORDEN );
    orderBy = "order by materia";
} else if ( frmPrincipal.getModuloActual().equals( PrincipalFrame.TIT_MODULO3 ) ) {
    sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.KARDEX_TODOS_SIN_ORDEN );
    orderBy = "order by matricula";
}

sql += " WHERE " + vecNombresColumnasBD.elementAt( jcboColumna.getSelectedIndex () );

String tipo    = vecTiposColumnas.elementAt ( jcboColumna.getSelectedIndex () );
String valor    = jtxfValor.getText ();

if ( jcboComparador.getSelectedItem().toString().equals ( "Sea igual a"      ) ) {
    sql += " = ? ";
} else if ( jcboComparador.getSelectedItem().toString().equals ( "Empiece con" ) ) {
    sql += " LIKE ? ";
    valor += "%";
} else if ( jcboComparador.getSelectedItem().toString().equals ( "Termine con" ) ) {
    sql += " LIKE ? ";
    valor = "%" + valor;
}

} else if ( jcboComparador.getSelectedItem().toString().equals ( "Contenga"      ) ) {
    sql += " LIKE ? ";
    valor = "%" + valor + "%";
} else if ( jcboComparador.getSelectedItem().toString().equals ( "Sea diferente" ) ) {
    sql += " <> ? ";
} else
    sql += " " + jcboComparador.getSelectedItem().toString () + " ? ";

sql += orderBy;

Object [][] args = { { tipo, valor } };
frmPrincipal.desplegarRegistros ( sql, args );
dispose ();
}
// -----

```

```

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(FiltrarDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
}

```

```

    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(FiltrarDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(FiltrarDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(FiltrarDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
}
//</editor-fold>

/* Create and display the dialog */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        FiltrarDialog dialog = new FiltrarDialog(new javax.swing.JFrame(), true);
        dialog.addWindowListener(new java.awt.event.WindowAdapter() {
            @Override
            public void windowClosing(java.awt.event.WindowEvent e) {
                System.exit(0);
            }
        });
        dialog.setVisible(true);
    }
});
}
}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JButton jButtonAceptar;
private javax.swing.JButton jButtonSalir;
private javax.swing.JComboBox<String> jcbColumna;
private javax.swing.JComboBox<String> jcbComparador;
private javax.swing.JTextField jtxfValor;
// End of variables declaration
}

```

Modulo1EditarDialog.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Dialogo que permite añadir y Editar información referente a los alumnos.
.*
.* Archivo   : Modulo1EditarDialog.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Dialogo que permite manejar la información de los Alumnos, tales como:
.*
.*          - Matricula
.*          - Nombre
.*          - Apellidos
.*          - Edad
.*          - Promedio

```


a través de su clase pertinente.

```

:* Ultima modif:
:* Fecha      Modific❖      Motivo
:* =====
=
:* 09/DIC/2020 Félix Mtz      Se añadió el dialog al proyecto y se le dió la definición básica..
   09/DIC/2020 Félix Mtz      Se añadieron los metodos pertinentes para acceder a la información.
   11/DIC/2020 Félix Mtz      Se corrigieron errores que evitaban que se añadiera información, como un valor no
actualizado.
:*-----*/
package app;

import java.sql.SQLException;
import java.util.Vector;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import modelo.Modelo1;
import mx.tecnm.itl.jdbc.EjecutorSQL;

public class Modulo1EditarDialog extends javax.swing.JDialog {

    private PrincipalFrame    frmPrincipal;
    private Modelo1           modelo;
    private String            accion;
    private Vector<String>     vecTiposColumnas;
    private String            sql;

    // -----
    public Modulo1EditarDialog ( java.awt.Frame parent, Modelo1 modelo ) {
        super ( parent, true );
        initComponents();

        frmPrincipal    = ( PrincipalFrame ) parent;
        this.modelo      = modelo;
        vecTiposColumnas = frmPrincipal.getVecTiposColumnas ();

        //    accion = ( modelo == null ) ? PrincipalFrame.NUEVO : PrincipalFrame.EDITAR;
        if ( modelo == null )
            accion = PrincipalFrame.NUEVO;
        else
            accion = PrincipalFrame.EDITAR;

        setTitle ( accion );

        inicializarFormulario ();
    }
    // -----
    private void inicializarFormulario () {
        if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
            jtxfMatricula.requestFocus ();
        } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
            jtxfMatricula.setText ( modelo.getMatricula () );
            jtxfNombre.setText ( modelo.getNombre () );
            jtxfApellidos.setText ( modelo.getApellidos () );
            jsnpEdad.setValue ( modelo.getEdad () );
            jftxfPromedio.setText ( modelo.getPromedio () + "" );
        }
    }
}

```

```

        jtxfNombre.requestFocus    ();
        jtxfMatricula.setEditable  ( false );
    }
}
// -----

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jtxfMatricula = new javax.swing.JTextField();
    jtxfNombre = new javax.swing.JTextField();
    jtxfApellidos = new javax.swing.JTextField();
    jspnEdad = new javax.swing.JSpinner();
    jftxfPromedio = new javax.swing.JFormattedTextField();
    jLabel6 = new javax.swing.JLabel();
    jbtnGuardar = new javax.swing.JButton();
    jbtnCancelar = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

    jLabel1.setText("Matricula:");

    jLabel2.setText("Nombre:");

    jLabel3.setText("Apellidos:");

    jLabel4.setText("Edad:");

    jLabel5.setText("Promedio:");

    jftxfPromedio.setHorizontalAlignment(javax.swing.JTextField.RIGHT);
    jftxfPromedio.setText("0.0");

    jLabel6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jLabel6.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/book_96px.png"))); // NOI18N
    jLabel6.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));

    jbtnGuardar.setText("Guardar");
    jbtnGuardar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jbtnGuardarActionPerformed(evt);
        }
    });

    jbtnCancelar.setText("Cancelar");
    jbtnCancelar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jbtnCancelarActionPerformed(evt);
        }
    });
}

```

```
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel3)
                .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 65,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jtxtfPromedio, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(26, 26, 26)
                        .addComponent(jbtnGuardar)
                        .addGap(99, 99, 99)
                        .addComponent(jbtnCancelar)))
                    .addGap(194, 194, 194))
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jtxtfNombre, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jtxtfMatricula, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jtxtfApellidos, javax.swing.GroupLayout.PREFERRED_SIZE, 210,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jspnEdad, javax.swing.GroupLayout.PREFERRED_SIZE, 60,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED_SIZE, 160,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(35, 35, 35)))
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(27, 27, 27)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jtxtfMatricula, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jLabel1)
                    .addGap(34, 34, 34)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel2)
                        .addComponent(jtxtfNombre, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
```

```

javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(jtxfApellidos, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
    .addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED_SIZE, 117,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(38, 38, 38)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel4)
        .addComponent(jspnEdad, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(35, 35, 35)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel5)
        .addComponent(jtxfPromedio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(18, 18, 18)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jbtnGuardar)
        .addComponent(jbtnCancelar))
    .addContainerGap(23, Short.MAX_VALUE)
);

pack();
setLocationRelativeTo(null);
} // </editor-fold>

private void jbtnCancelarActionPerformed(java.awt.event.ActionEvent evt) {
    dispose ();
}

// -----
private void jbtnGuardarActionPerformed(java.awt.event.ActionEvent evt) {
    Object[][] args = null;
    String mensaje = "";

    if ( validarDatos () == false )
        return;

    // Se determina cual sentencia SQL se utilizará y se forma la matriz de argumentos
    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
        mensaje = "El registro ha sido agregado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.ALUMNOS_INSERTA_NUEVO );

        args = new Object[][] {
            { vecTiposColumnas.elementAt ( 0 ), modelo.getMatricula () },
            { vecTiposColumnas.elementAt ( 1 ), modelo.getNombre () },
            { vecTiposColumnas.elementAt ( 2 ), modelo.getApellidos () },
            { vecTiposColumnas.elementAt ( 3 ), modelo.getEdad () },
            { vecTiposColumnas.elementAt ( 4 ), modelo.getPromedio () }
        };
    }
    else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        mensaje = "El registro ha sido modificado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.ALUMNOS_ACTUALIZA_DATOS );
        args = new Object [][] {
            { vecTiposColumnas.elementAt ( 1 ), modelo.getNombre () },
            { vecTiposColumnas.elementAt ( 2 ), modelo.getApellidos () },
            { vecTiposColumnas.elementAt ( 3 ), modelo.getEdad () },

```

```
{ vecTiposColumnas.elementAt ( 4 ), modelo.getPromedio () },
{ vecTiposColumnas.elementAt ( 0 ), modelo.getMatricula () }
};
}

try {
    int regs = EjecutorSQL.sqlEjecutar ( sql, args );
    if ( regs == 1 ) {
        frmPrincipal.getJbtnModulo1().doClick ();
        JOptionPane.showMessageDialog( this,
            mensaje,
            accion,
            JOptionPane.INFORMATION_MESSAGE );
    }
} catch ( SQLException ex ) {
    dialogoMensaje ( ex.toString () );
}

dispose ();

}
// -----
private boolean validarDatos () {
    String matricula = jtxfMatricula.getText ();
    if ( matricula.trim().equals ( "" ) ) {
        dialogoMensaje ( "No se permite un valor en blanco" );
        jtxfMatricula.requestFocus ();
        return false;
    }

    String nombre = jtxfNombre.getText ();
    if ( nombre.trim().equals ( "" ) ) {
        dialogoMensaje ( "No se permite un valor en blanco" );
        jtxfNombre.requestFocus ();
        return false;
    }

    String apellidos = jtxfApellidos.getText ();
    if ( apellidos.trim().equals ( "" ) ) {
        dialogoMensaje ( "No se permite un valor en blanco" );
        jtxfApellidos.requestFocus ();
        return false;
    }

    int edad = 0;
    try {
        edad = Integer.parseInt ( jspnEdad.getValue().toString () );
    } catch ( NumberFormatException e ) {
        dialogoMensaje ( "Debe capturar un valor numérico válido" );
        jspnEdad.requestFocus ();
        return false;
    }

    float promedio;
    try {
        promedio = Float.parseFloat ( jftxfPromedio.getText () );
    } catch ( NumberFormatException e ) {
        dialogoMensaje ( "Debe capturar un valor numérico válido" );
    }
}
```

```

        jspnEdad.requestFocus ();
        return false;
    }

    modelo = new Modelo1 ( matricula, nombre, apellidos, edad, promedio );
    return true;
}
// -----
private void dialogoMensaje ( String mensaje ) {
    JOptionPane.showMessageDialog( this, mensaje, "Error", JOptionPane.ERROR_MESSAGE );
}
// -----

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(Modulo1EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Modulo1EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(Modulo1EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(Modulo1EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    }
}
//</editor-fold>

    /* Create and display the dialog */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            Modulo1EditarDialog dialog = new Modulo1EditarDialog(new javax.swing.JFrame(), null );
            dialog.addWindowListener(new java.awt.event.WindowAdapter() {
                @Override
                public void windowClosing(java.awt.event.WindowEvent e) {
                    System.exit(0);
                }
            });
            dialog.setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;

```

```

private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JButton jbbtnCancelar;
private javax.swing.JButton jbbtnGuardar;
private javax.swing.JFormattedTextField jftxfPromedio;
private javax.swing.JSpinner jspnEdad;
private javax.swing.JTextField jtxfApellidos;
private javax.swing.JTextField jtxfMatricula;
private javax.swing.JTextField jtxfNombre;
// End of variables declaration
}

```

Modulo2EditarDialog.java

```

/*-----
.*          TECNOLOGICO NACIONAL DE MEXICO
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: AGO-DIC/2020   HORA: 17-18 HRS
.*
.*          Dialogo que permite añadir y Editar información referente a las Materias.
.*
.* Archivo   : Modulo2EditarDialog.java
.* Autor     : Félix Gerardo Martínez Hinojo   17130800
.* Fecha     : 01/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripción : Dialogo que permite manejar la información de las Materias, tales como:
.*
.*          - ID Materia
.*          - Materia
.*          - Creditos
.*          a través de su clase pertinente.
.* Ultima modif:
.* Fecha      Modific*      Motivo
.*=====
.* 11/DIC/2020 Félix Mtz   Se añadió el dialog al proyecto y se le dió la definición básica..
.* 11/DIC/2020 Félix Mtz   Se añadieron los metodos pertinentes para acceder a la información.
.* 14/DIC/2020 Félix Mtz   Se corrigieron errores que evitaban que se añadiera información, como un valor no
.* actualizado.
.*-----*/
package app;

import java.sql.SQLException;
import java.util.Vector;
import javax.swing.JOptionPane;
import modelo.Modelo1;
import modelo.Modelo2;
import mx.tecnm.itl.jdbc.EjecutorSQL;

public class Modulo2EditarDialog extends javax.swing.JDialog {

    private PrincipalFrame    frmPrincipal;
    private Modelo2           modelo;

```

```

private String      accion;
private Vector<String>  vecTiposColumnas;
private String      sql;

// -----
public Modulo2EditarDialog(java.awt.Frame parent, Modelo2 modelo ) {
    super(parent, true );
    initComponents();

    frmPrincipal      = ( PrincipalFrame ) parent;
    this.modelo        = modelo;
    vecTiposColumnas  = frmPrincipal.getVecTiposColumnas ();

//    accion = ( modelo == null ) ? PrincipalFrame.NUEVO : PrincipalFrame.EDITAR;
    if ( modelo == null )
        accion = PrincipalFrame.NUEVO;
    else
        accion = PrincipalFrame.EDITAR;

    setTitle ( accion );

    inicializarFormulario ();
}

// -----

private void inicializarFormulario () {
    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
        jtxfIDMateria.requestFocus ();
    } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        jtxfIDMateria.setText      ( modelo.getIdmateria  () );
        jtxfMateria.setText        ( modelo.getMateria    () );
        jtxfCreditos.setText       ( modelo.getCreditos   () + "" );

        jtxfMateria.requestFocus  ();
        jtxfIDMateria.setEditable  ( false );
    }
}

// -----

/**
 * 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() {

    jLabel1 = new javax.swing.JLabel();
    jtxfIDMateria = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    jtxfMateria = new javax.swing.JTextField();
    jLabel3 = new javax.swing.JLabel();
    jtxfCreditos = new javax.swing.JTextField();
    jLabel6 = new javax.swing.JLabel();
    btnCancelar = new javax.swing.JButton();
    btnGuardar_ = new javax.swing.JButton();

```



```
setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

jLabel1.setText("ID Materia:");

jLabel2.setText("Materia:");

jLabel3.setText("Creditos:");

jLabel6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel6.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/Student_96px.png"))); // NOI18N
jLabel6.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));

jbtnCancelar.setText("Cancelar");
jbtnCancelar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnCancelarActionPerformed(evt);
    }
});

jbtnGuardar_.setText("Guardar");
jbtnGuardar_.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jbtnGuardar_ActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(122, 122, 122)
            .addComponent(jbtnGuardar_)
            .addGap(129, 129, 129)
            .addComponent(jbtnCancelar)
            .addGap(138, Short.MAX_VALUE))
        .addGroup(layout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 47, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel3)
                .addGap(18, 18, 18)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jtxfMateria, javax.swing.GroupLayout.PREFERRED_SIZE, 210, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jtxfIDMateria, javax.swing.GroupLayout.PREFERRED_SIZE, 210, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jtxfCreditos, javax.swing.GroupLayout.PREFERRED_SIZE, 210, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED_SIZE, 160, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(46, 46, 46))
            .addGap(46, 46, 46))
);
```

```

);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(28, 28, 28)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jtxfldMateria, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jLabel1)
                    .addGap(34, 34, 34)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel2)
                        .addComponent(jtxfMateria, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel3)
                        .addComponent(jtxfCreditos, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
                    .addComponent(jLabel6, javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 117, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(50, 50, 50)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jbtnCancelar)
                    .addComponent(jbtnGuardar_))
                .addContainerGap(43, Short.MAX_VALUE))
        );

pack();
setLocationRelativeTo(null);
} // </editor-fold>
// -----
private void jbtnCancelarActionPerformed(java.awt.event.ActionEvent evt) {
    dispose ();
}
// -----
private void jbtnGuardar_ActionPerformed(java.awt.event.ActionEvent evt) {
    Object[][] args = null;
    String mensaje = "";

    if ( validarDatos () == false )
        return;

    // Se determina cual sentencia SQL se utilizará y se forma la matriz de argumentos
    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
        mensaje = "El registro ha sido agregado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.MATERIAS_INSERTA_NUEVO );

        args = new Object[][] {
            { vecTiposColumnas.elementAt ( 0 ), modelo.getIdmateria () },
            { vecTiposColumnas.elementAt ( 1 ), modelo.getMateria () },
            { vecTiposColumnas.elementAt ( 2 ), modelo.getCreditos () }
        };
    } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        mensaje = "El registro ha sido modificado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.MATERIAS_ACTUALIZA_DATOS );
    }
}

```

```

        args = new Object [][] {
            { vecTiposColumnas.elementAt ( 1 ), modelo.getMateria      () },
            { vecTiposColumnas.elementAt ( 2 ), modelo.getCreditos     () },
            { vecTiposColumnas.elementAt ( 0 ), modelo.getIdmateria     () }
        };
    }

    try {
        int regs = EjecutorSQL.sqlEjecutar ( sql, args );
        if ( regs == 1 ) {
            frmPrincipal.getJbtnModulo2().doClick ();
            JOptionPane.showMessageDialog( this,
                mensaje,
                accion,
                JOptionPane.INFORMATION_MESSAGE );
        }
    } catch ( SQLException ex ) {
        dialogoMensaje ( ex.toString () );
    }

    dispose ();
}
// -----
private boolean validarDatos () {
    String idmateria = jtxfIDMateria.getText ();
    if ( idmateria.trim().equals ( "" ) ) {
        dialogoMensaje ( "No se permite un valor en blanco" );
        jtxfIDMateria.requestFocus ();
        return false;
    }

    String materia = jtxfMateria.getText ();
    if ( materia.trim().equals ( "" ) ) {
        dialogoMensaje ( "No se permite un valor en blanco" );
        jtxfMateria.requestFocus ();
        return false;
    }

    int creditos;
    try {
        creditos = Integer.parseInt ( jtxfCreditos.getText () );
    } catch ( NumberFormatException e ) {
        dialogoMensaje ( "Debe capturar un valor numérico válido" );
        jtxfCreditos.requestFocus ();
        return false;
    }

    modelo = new Modelo2 ( idmateria, materia, creditos );
    return true;
}
// -----
private void dialogoMensaje ( String mensaje ) {
    JOptionPane.showMessageDialog( this, mensaje, "Error", JOptionPane.ERROR_MESSAGE );
}
// -----

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(Modulo2EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(Modulo2EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
} catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(Modulo2EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
    java.util.logging.Logger.getLogger(Modulo2EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
}
//</editor-fold>

/* Create and display the dialog */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        Modulo2EditarDialog dialog = new Modulo2EditarDialog(new javax.swing.JFrame(), null );
        dialog.addWindowListener(new java.awt.event.WindowAdapter() {
            @Override
            public void windowClosing(java.awt.event.WindowEvent e) {
                System.exit(0);
            }
        });
        dialog.setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel6;
private javax.swing.JButton jButtonCancelar;
private javax.swing.JButton jButtonGuardar_;
private javax.swing.JTextField jtxfCreditos;
private javax.swing.JTextField jtxfIDMateria;
private javax.swing.JTextField jtxfMateria;
// End of variables declaration
}

```

Modulo3EditarDialog.java

```

/*-----
:*          TECNOLOGICO NACIONAL DE MEXICO
:*          INSTITUTO TECNOLOGICO DE LA LAGUNA

```

INGENIERIA EN SISTEMAS COMPUTACIONALES
TOPICOS AVANZADOS DE PROGRAMACION "B"

SEMESTRE: AGO-DIC/2020 HORA: 17-18 HRS

Dialogo que permite añadir y Editar información referente al Kardex de los alumnos.

* Archivo : Modulo3EditarDialog.java
* Autor : Félix Gerardo Martínez Hinojo 17130800
* Fecha : 01/DIC/2020
* Compilador : JAVA J2SE v1.8.0
* Descripción : Dialogo que permite manejar la información del Kardex de los alumnos, tales como:

- ID Calificación
- Nombre Alumno
- Apellidos Alumno
- ID Materia
- Materia
- entre otras.

a través de su clase pertinente y relaciones.

* Última modif:

* Fecha Modificación Motivo

*=====

=

* 14/DIC/2020 Félix Mtz Se añadió el dialog al proyecto y se le dió la definición básica..
14/DIC/2020 Félix Mtz Se añadieron los metodos pertinentes para acceder a la información.
15/DIC/2020 Félix Mtz Se corrigieron errores que evitaban que se añadiera información, como un valor no actualizado.

-----/

package app;

```
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Vector;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import modelo.Modelo1;
import modelo.Modelo2;
import modelo.Modelo3;
import mx.tecnm.itl.jdbc.EjecutorSQL;
import mx.tecnm.itl.util.Imagenes;
```

```
public class Modulo3EditarDialog extends javax.swing.JDialog {
```

```
    private PrincipalFrame frmPrincipal;
    private Modelo3 modelo;
    private String accion;
    private Vector<String> vecTiposColumnas;
    private String sql;
```

```
// -----
```

```
public Modulo3EditarDialog(java.awt.Frame parent, Modelo3 modelo ) {
    super ( parent, true );
    initComponents();
```

```
    frmPrincipal = ( PrincipalFrame ) parent;
```

```

this.modelo      = modelo;
vecTiposColumnas = frmPrincipal.getVecTiposColumnas ();

//  accion = ( modelo == null ) ? PrincipalFrame.NUEVO : PrincipalFrame.EDITAR;
if ( modelo == null )
    accion = PrincipalFrame.NUEVO;
else
    accion = PrincipalFrame.EDITAR;

setTitle ( accion );

//  jlblLogo.setICon ( Imagenes.escalarImagen ( jlblLogo.getICon (), 105, 105 ) );

llenarCombo1DesdeTabla ();
llenarCombo2DesdeTabla ();
inicializarFormulario ();
}
// -----
private void llenarCombo1DesdeTabla () {
    ResultSet rs;
    Modelo1 modelo1;

    // Llenado del combo box de alumnos
    sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.ALUMNOS_TODOS_POR_NOMBRE );

    try {
        rs = EjecutorSQL.sqlQuery ( sql );
        jcboAlumno.addItem ( new Modelo1 ( "", "--Seleccione--", "", 0, 0.0f ) );
        while ( rs.next () ) {
            String matricula = rs.getString ( "matricula" );
            String nombre = rs.getString ( "nombre" );
            String apellidos = rs.getString ( "apellidos" );

            modelo1 = new Modelo1 ( matricula, nombre, apellidos, 0, 0.0f );
            jcboAlumno.addItem ( modelo1 );

            if ( modelo != null && matricula.equals ( modelo.getMatricula () ) ) {
                int index = jcboAlumno.getModel ().getSize();
                jcboAlumno.setSelectedIndex ( index - 1 );
            }
        }
        rs.close ();
    } catch ( SQLException ex ) {
        dialogoMensaje( ex.toString () );
    }
}
// -----
private void llenarCombo2DesdeTabla () {
    ResultSet rs;
    Modelo2 modelo2;

    // Llenado del combo box de alumnos
    sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.MATERIAS_TODOS_POR_MATERIA );

    try {
        rs = EjecutorSQL.sqlQuery ( sql );
        jcboMateria.addItem ( new Modelo2 ( "", "--Seleccione--", 0 ) );
    }
}

```

```

while ( rs.next () ) {
    String idmateria = rs.getString ( "idmateria" );
    String materia = rs.getString ( "materia" );

    modelo2 = new Modelo2 ( idmateria, materia, 0 );
    jcboMateria.addItem ( modelo2 );

    if ( modelo != null && idmateria.equals ( modelo.getIdmateria () ) ) {
        int index = jcboMateria.getModel ().getSize();
        jcboMateria.setSelectedIndex ( index - 1 );
    }
}
rs.close ();

} catch ( SQLException ex ) {
    dialogoMensaje( ex.toString () );
}
}
// -----
private void inicializarFormulario () {
    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {

    } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        jcboPeriodo.setSelectedItem ( modelo.getPeriodo () );
        jtxfCalific.setText ( modelo.getCalificacion () + "" );
    }
    jcboAlumno.requestFocus ();
}
// -----

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    btnGuardar_ = new javax.swing.JButton();
    btnCancelar = new javax.swing.JButton();
    lblLogo = new javax.swing.JLabel();
    jcboAlumno = new javax.swing.JComboBox<>();
    jcboMateria = new javax.swing.JComboBox<>();
    jcboPeriodo = new javax.swing.JComboBox<>();
    jLabel4 = new javax.swing.JLabel();
    jtxfCalific = new javax.swing.JFormattedTextField();

    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

    jLabel1.setText("Alumno:");

    jLabel2.setText("Materia");

    jLabel3.setText("Periodo");

    btnGuardar_.setText("Guardar");
    btnGuardar_.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnGuardar_ActionPerformed(evt);

```

```

    }
    });

    jbtnCancelar.setText("Cancelar");
    jbtnCancelar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jbtnCancelarActionPerformed(evt);
        }
    });

    jlblLogo.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    jlblLogo.setIcon(new javax.swing.ImageIcon(getClass().getResource("/iconos/kardex_96px_1.png"))); // NOI18N
    jlblLogo.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));

    jcboPeriodo.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "2018A", "2018B", "2019A",
"2019B", "2020A", "2020B", "2021A", "2021B", "2022A", "2022B" }));

    jLabel4.setText("Calificación");

    jtxfCalific.setFormatterFactory(new javax.swing.text.DefaultFormatterFactory(new
javax.swing.text.NumberFormatter(new java.text.DecimalFormat("#0"))));
    jtxfCalific.setHorizontalAlignment(javax.swing.JTextField.RIGHT);

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(45, 45, 45)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jLabel1)
                    .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel3)
                    .addComponent(jLabel4))
                .addGap(34, 34, 34)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(jbtnGuardar_)
                        .addGap(95, 95, 95)
                        .addComponent(jbtnCancelar)
                        .addGap(0, 0, Short.MAX_VALUE))
                    .addGroup(layout.createSequentialGroup()
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(layout.createSequentialGroup()
                                .addComponent(jcboPeriodo, javax.swing.GroupLayout.PREFERRED_SIZE, 206,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent(jcboMateria, javax.swing.GroupLayout.PREFERRED_SIZE, 206,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent(jcboAlumno, javax.swing.GroupLayout.PREFERRED_SIZE, 206,
javax.swing.GroupLayout.PREFERRED_SIZE))
                            .addGap(67, 67, 67)
                            .addComponent(jlblLogo, javax.swing.GroupLayout.PREFERRED_SIZE, 160,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(layout.createSequentialGroup()
                                .addComponent(jtxfCalific, javax.swing.GroupLayout.PREFERRED_SIZE, 105,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addGap(93, 93, Short.MAX_VALUE))))
                    .addGap(93, 93, Short.MAX_VALUE))))
    );

```



```

);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(28, 28, 28)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGroup(layout.createSequentialGroup()
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel1)
                        .addComponent(jcboAlumno, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGap(18, 18, 18)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel2)
                        .addComponent(jcboMateria, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGap(18, 18, 18)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel3)
                        .addComponent(jcboPeriodo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addComponent(jlblLogo, javax.swing.GroupLayout.PREFERRED_SIZE, 117,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(27, 27, 27)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel4)
                    .addComponent(jtxfCalific, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 20, Short.MAX_VALUE)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jbtnCancelar)
                    .addComponent(jbtnGuardar_))
                .addGap(50, 50, 50))
        );

pack();
setLocationRelativeTo(null);
} // </editor-fold>
// -----
private void jbtnGuardar_ActionPerformed(java.awt.event.ActionEvent evt) {
    Object[][] args = null;
    String mensaje = "";

    if ( validarDatos () == false )
        return;

    // Se determina cual sentencia SQL se utilizará y se forma la matriz de argumentos
    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
        mensaje = "El registro ha sido agregado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.KARDEX_INSERTA_NUEVO );

        args = new Object[][] {
            { vecTiposColumnas.elementAt ( 0 ), modelo.getIdcalific      () },
            { vecTiposColumnas.elementAt ( 1 ), modelo.getMatricula      () },
            { vecTiposColumnas.elementAt ( 4 ), modelo.getIdmateria      () },
            { vecTiposColumnas.elementAt ( 6 ), modelo.getPeriodo        () },
            { vecTiposColumnas.elementAt ( 7 ), modelo.getCalificacion    () }
        };
    }

```

```

    } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        mensaje = "El registro ha sido modificado.";
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.KARDEX_ACTUALIZA_DATOS );
        args = new Object [][] {
            { vecTiposColumnas.elementAt ( 1 ), modelo.getMatricula      () },
            { vecTiposColumnas.elementAt ( 4 ), modelo.getIdmateria      () },
            { vecTiposColumnas.elementAt ( 6 ), modelo.getPeriodo        () },
            { vecTiposColumnas.elementAt ( 7 ), modelo.getCalificacion    () },
            { vecTiposColumnas.elementAt ( 0 ), modelo.getIdcalific      () }
        };
    }

    try {
        int regs = EjecutorSQL.sqlEjecutar ( sql, args );
        if ( regs == 1 ) {
            frmPrincipal.getJbtnModulo3().doClick ();
            JOptionPane.showMessageDialog( this,
                mensaje,
                accion,
                JOptionPane.INFORMATION_MESSAGE );
        }
    } catch ( SQLException ex ) {
        dialogoMensaje ( ex.toString () );
    }

    dispose ();
}
// -----
private void jbtnCancelarActionPerformed(java.awt.event.ActionEvent evt) {
    dispose ();
}
// -----
private boolean validarDatos ( ) {
    Modelo1 modelo1 = (Modelo1) jcboAlumno.getSelectedItem ();
    String matricula = modelo1.getMatricula ();
    String nombre = modelo1.getNombre ();

    if ( nombre.equals ( "--Seleccione--" ) ) {
        dialogoMensaje ( "Debe seleccionar un elemento de la lista" );
        jcboAlumno.requestFocus ();
        return false;
    }

    Modelo2 modelo2 = (Modelo2) jcboMateria.getSelectedItem ();
    String idmateria = modelo2.getIdmateria ();
    String materia = modelo2.getMateria ();

    if ( materia.equals ( "--Seleccione--" ) ) {
        dialogoMensaje ( "Debe seleccionar un elemento de la lista" );
        jcboMateria.requestFocus ();
        return false;
    }

    int calificacion = 0;
    try {
        calificacion = Integer.parseInt ( jtxfCalific.getText () );
    } catch ( NumberFormatException e ) {
        dialogoMensaje ( "Debe capturar un valor numérico válido" );
    }
}

```

```

        return false;
    }

    String periodo = jcboPeriodo.getSelectedItem ().toString ();
    int idcalific = 0;

    if ( accion.equals ( PrincipalFrame.NUEVO ) ) {
        ResultSet rs;

        // DETERMINAR EL CONSECUTIVO DE ID CALIFICACION E INCREMENTARLO EN 1
        sql = frmPrincipal.getPropConsultasSQL().getProperty ( PrincipalFrame.KARDEX_OBT_MAX_ID );
        try {
            rs = EjecutorSQL.sqlQuery ( sql );
            if ( rs.next () ) {
                idcalific = rs.getInt ( "maxid" );
                idcalific++;
            }
        } catch ( SQLException ex ) {
            dialogoMensaje ( ex.toString () );
        }
    } else if ( accion.equals ( PrincipalFrame.EDITAR ) ) {
        idcalific = modelo.getIdcalific ();
    }

    modelo = new Modelo3 ( idcalific, matricula, idmateria, periodo, calificacion );
    return true;
}

// -----
private void dialogoMensaje ( String mensaje ) {
    JOptionPane.showMessageDialog( this, mensaje, "Error", JOptionPane.ERROR_MESSAGE );
}

// -----
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(Modulo3EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Modulo3EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(Modulo3EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(Modulo3EditarDialog.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    }
}

```

//</editor-fold>

/* Create and display the dialog */

```

java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        Modulo3EditarDialog dialog = new Modulo3EditarDialog(new javax.swing.JFrame(), null );
        dialog.addWindowListener(new java.awt.event.WindowAdapter() {
            @Override
            public void windowClosing(java.awt.event.WindowEvent e) {
                System.exit(0);
            }
        });
        dialog.setVisible(true);
    }
});
}
}

```

// Variables declaration - do not modify

```

private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JButton jButtonCancelar;
private javax.swing.JButton jButtonGuardar_;
private javax.swing.JComboBox<Modelo1> jcboAlumno;
private javax.swing.JComboBox<Modelo2> jcboMateria;
private javax.swing.JComboBox<String> jcboPeriodo;
private javax.swing.JLabel lblLogo;
private javax.swing.JFormattedTextField jtxfCalific;
// End of variables declaration

```

}

AcercaDeDialog.java

```

/*-----
.*          INSTITUTO TECNOLOGICO DE LA LAGUNA
.*          INGENIERIA EN SISTEMAS COMPUTACIONALES
.*          TOPICOS AVANZADOS DE PROGRAMACION "B"
.*
.*          SEMESTRE: ENE-JUN/2020   HORA: 17-18 HRS
.*
.*          Clase jAcercaDe
.*
.* Archivo   : jAcercaDeDialog.java
.* Autor     : Félix Gerardo Martínez Hinojo 17130800
.* Fecha     : 17/DIC/2020
.* Compilador : JAVA J2SE v1.8.0
.* Descripcion : Este es un Dialog que carga un bean que muestra la información referente a la aplicación. Permite
.*              editarla también mediante
.*              las propiedades del Dialog y del Bean, entrelazandolos.
.*
.* Ultima modif:
.* Fecha      Modific❖      Motivo
.*=====
.* 17/DIC/2020 Félix Mtz      Agregar prologo.
.*-----*/
package app;

```

```

public class jAcercaDeDialog extends javax.swing.JDialog {

    // -----
    public jAcercaDeDialog(java.awt.Frame parent, boolean modal) {
        super(parent, modal);
        initComponents();
    }
    // -----

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jAcercaDe1 = new mx.tecnm.itl.beans.jAcercaDe();
        jAcercaDe2 = new mx.tecnm.itl.beans.jAcercaDe();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

        jAcercaDe2.setCarrera("Ing. Sistemas Computacionales");
        jAcercaDe2.setDesarrollador("Félix Gerardo Martínez Hinojo");
        jAcercaDe2.setDesarrollador2("");
        jAcercaDe2.setDesarrollador3("");
        jAcercaDe2.setDesarrollador4("");
        jAcercaDe2.setInstitucion("Tecnológico Nacional de México");
        jAcercaDe2.setMateria("Tópicos Avanzados De Programación");
        jAcercaDe2.setNombre("Acceso a Bases de datos");
        jAcercaDe2.setVersion("1.0.0");
        jAcercaDe2.setUniversidad("Instituto Tecnológico De La Laguna");

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .add(jAcercaDe2, javax.swing.GroupLayout.PREFERRED_SIZE, 705,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(0, 10, Short.MAX_VALUE))
                    .addContainerGap())
                .addGroup(layout.createSequentialGroup()
                    .add(jAcercaDe2, javax.swing.GroupLayout.PREFERRED_SIZE, 311,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addContainerGap())
        );

        pack();
        setLocationRelativeTo(null);
    } // </editor-fold>

    // -----

    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

```

```

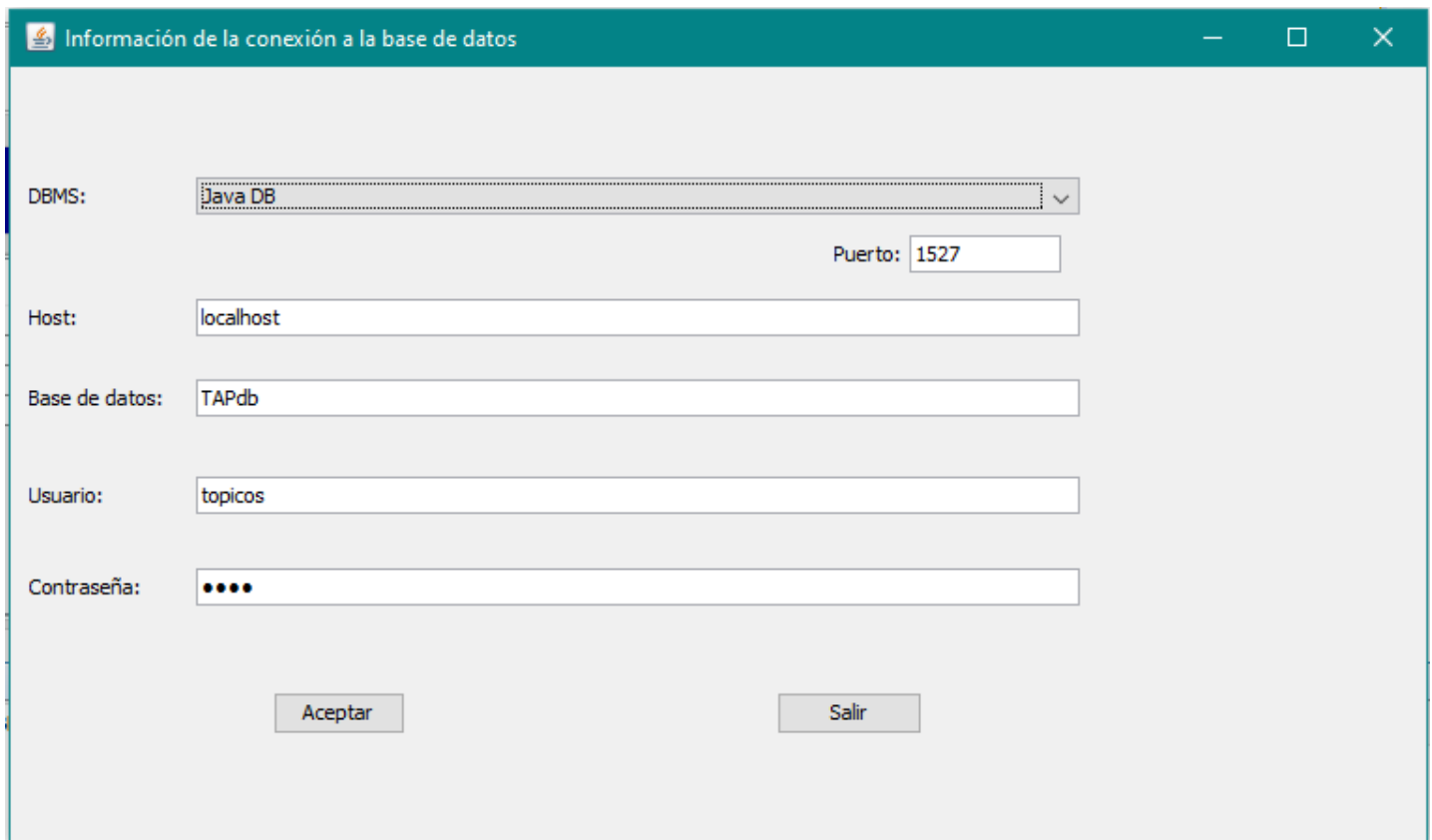
/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
 * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
 */
try {
    for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(jAcercaDeDialog.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(jAcercaDeDialog.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(jAcercaDeDialog.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
    java.util.logging.Logger.getLogger(jAcercaDeDialog.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
}
//</editor-fold>

/* Create and display the dialog */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        jAcercaDeDialog dialog = new jAcercaDeDialog(new javax.swing.JFrame(), true);
        dialog.addWindowListener(new java.awt.event.WindowAdapter() {
            @Override
            public void windowClosing(java.awt.event.WindowEvent e) {
                System.exit(0);
            }
        });
        dialog.setVisible(true);
    }
});
}

// Variables declaration - do not modify
private mx.tecnm.itl.beans.jAcercaDe jAcercaDe1;
private mx.tecnm.itl.beans.jAcercaDe jAcercaDe2;
// End of variables declaration
}

```

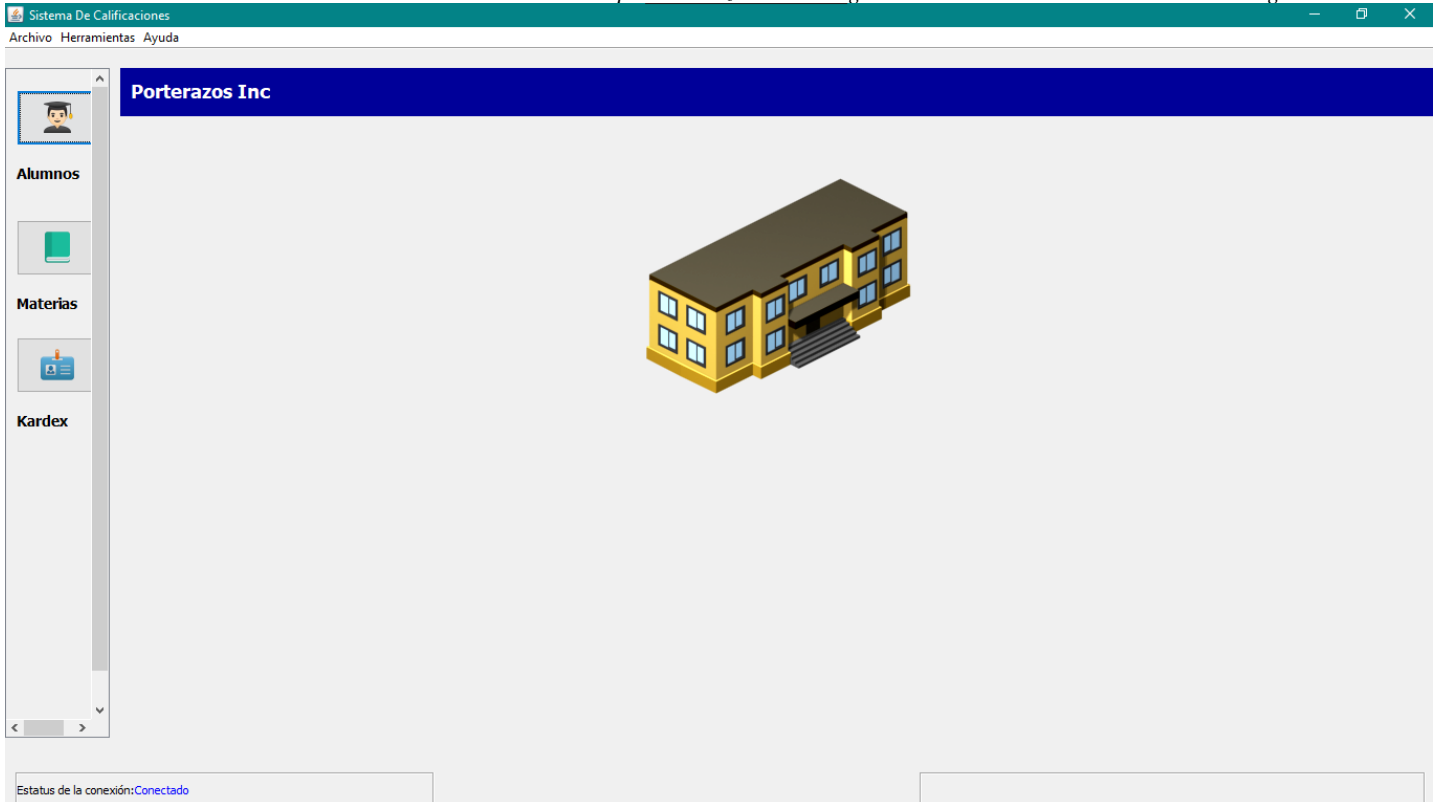
Prueba de Ejecución



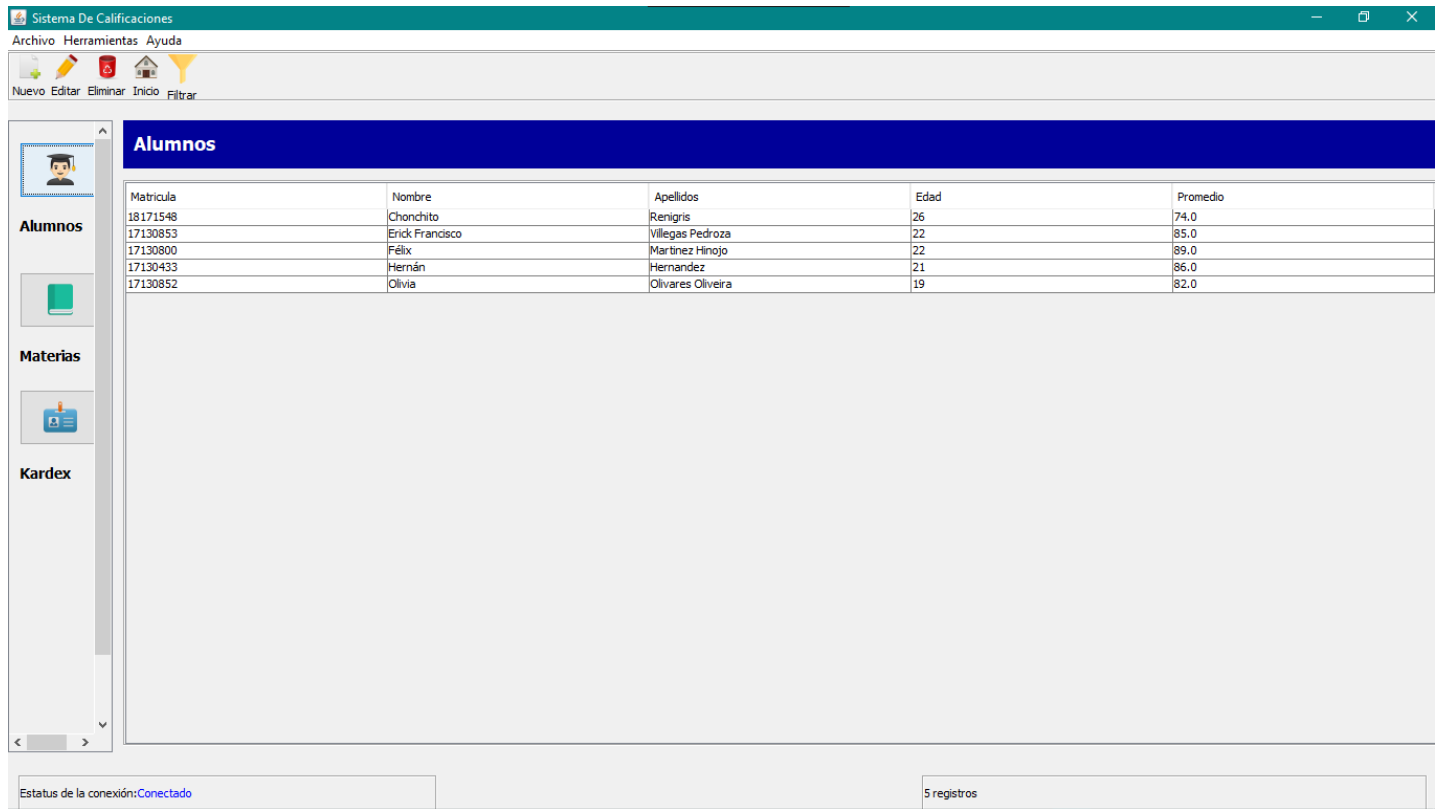
The screenshot shows a window titled "Información de la conexión a la base de datos" with a teal header bar. Inside the window, there are several input fields and buttons. The "DBMS:" field is a dropdown menu with "Java DB" selected. To its right is a "Puerto:" field with the value "1527". Below these are text input fields for "Host:" (containing "localhost"), "Base de datos:" (containing "TAPdb"), "Usuario:" (containing "topicos"), and "Contraseña:" (containing four dots). At the bottom, there are two buttons: "Aceptar" and "Salir".

Field	Value
DBMS:	Java DB
Puerto:	1527
Host:	localhost
Base de datos:	TAPdb
Usuario:	topicos
Contraseña:	••••

Tenemos un menú en el cual podremos elegir la base de datos a la cual accederemos, en este caso ya viene por default la información de la base de datos con la cual trabajaremos.



Tenemos un menú de inicio en el cual podremos acceder a todas las funciones de la aplicación

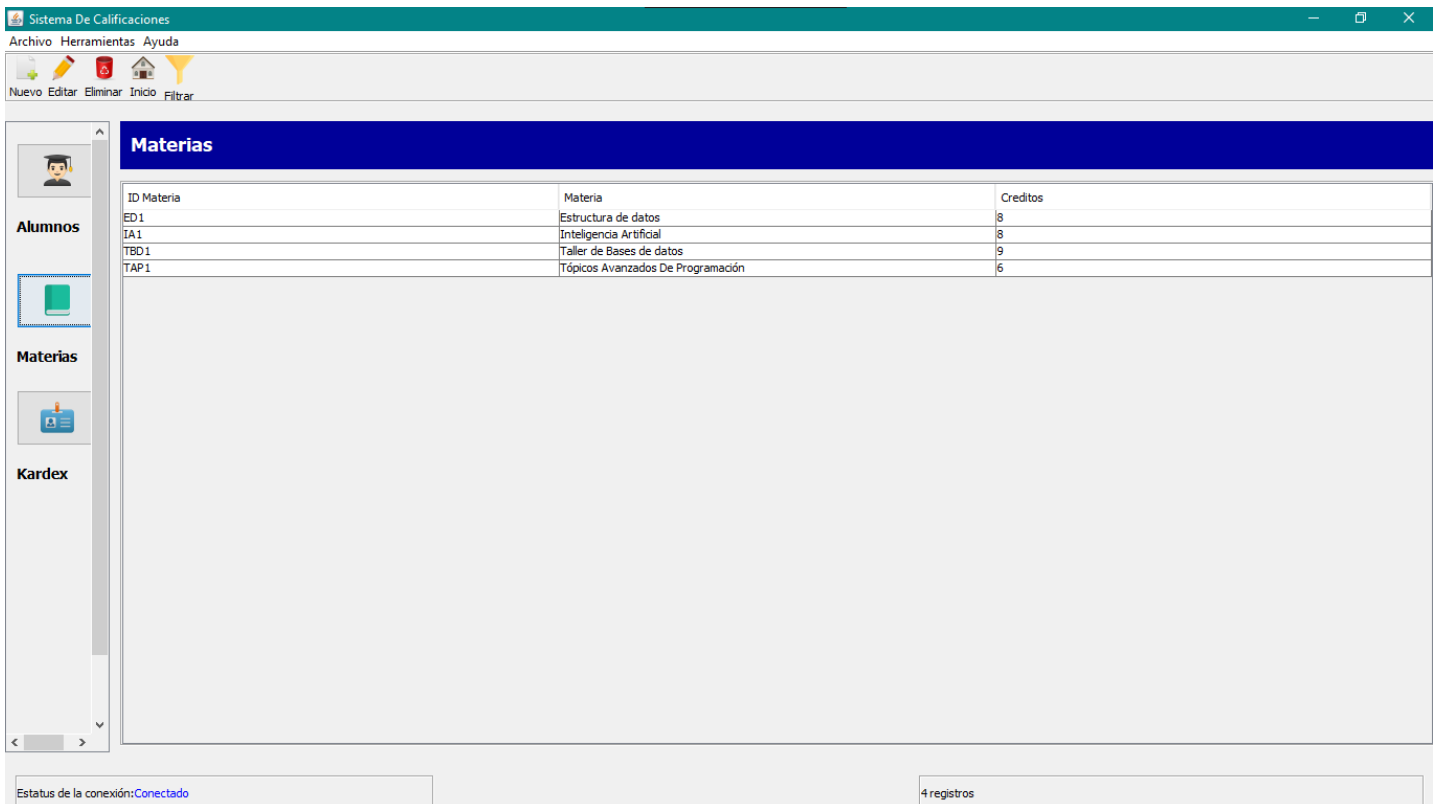


Matricula	Nombre	Apellidos	Edad	Promedio
18171548	Chonchito	Renigris	26	74.0
17130853	Erick Francisco	Villegas Pedroza	22	85.0
17130800	Félix	Martínez Hinojo	22	89.0
17130433	Hernán	Hernandez	21	86.0
17130852	Olivia	Olivares Oliveira	19	82.0

Estatus de la conexión: Conectado

5 registros

Al dar clic al botón Alumnos se desplegará el menú (Alumnos), el cual nos presenta los datos de los alumnos como matrícula, nombre, apellidos, edad y promedio general.



ID Materia	Materia	Creditos
ED 1	Estructura de datos	8
IA 1	Inteligencia Artificial	8
TBD 1	Taller de Bases de datos	9
TAP 1	Tópicos Avanzados De Programación	6


Estatus de la conexión: Conectado


4 registros


Al dar clic al botón Materias se desplegará el menú (Materias), el cual nos presenta los atributos de las materias como ID materia, nombre de la materia y la cantidad de créditos los cuales la materia proporciona al acreditarla.

Sistema De Calificaciones							
Archivo Herramientas Ayuda							
Nuevo Editar Eliminar Inicio Filtrar							
Kardex							
ID Calific	Matricula	Nombre	Apellidos	ID Materia	Materia	Periodo	Calificación
8	17130433	Hernán	Hernandez	IA1	Inteligencia Artificial	2020A	91
7	17130433	Hernán	Hernandez	TAP1	Tópicos Avanzados De Progr...	2020A	95
6	17130433	Hernán	Hernandez	ED1	Estructura de datos	2020A	81
5	17130433	Hernán	Hernandez	TBD1	Taller de Bases de datos	2020A	78
12	17130800	Félix	Martinez Hinojo	IA1	Inteligencia Artificial	2020A	90
11	17130800	Félix	Martinez Hinojo	TAP1	Tópicos Avanzados De Progr...	2020A	94
10	17130800	Félix	Martinez Hinojo	ED1	Estructura de datos	2020A	86
9	17130800	Félix	Martinez Hinojo	TBD1	Taller de Bases de datos	2020A	88
3	17130852	Olivia	Oliveras Oliveira	TAP1	Tópicos Avanzados De Progr...	2020A	81
2	17130852	Olivia	Oliveras Oliveira	ED1	Estructura de datos	2020A	75
1	17130852	Olivia	Oliveras Oliveira	TBD1	Taller de Bases de datos	2020A	91
16	17130853	Erick Francisco	Villegas Pedroza	ED1	Estructura de datos	2020B	100
15	17130853	Erick Francisco	Villegas Pedroza	TAP1	Tópicos Avanzados De Progr...	2020B	70
13	18171548	Chonchito	Renigris	ED1	Estructura de datos	2020A	74

Al dar clic al botón Kardex se desplegará el menú (Kardex), el cual nos presenta los datos de todos los alumnos y de las materias que han acreditado, como ID Calif, Matrícula, Nombre y Apellidos del Alumno, ID materia, Nombre de la Materia, Periodo en el cual el Alumno tomo esa materia y la calificación obtenida.

**Alumnos**

**Materias**

**Kardex**

Alumnos

Matricula	Nombre	Apellidos	Edad	Promedio
18171548	Chonchito	Benitois	16	74.0
17130853	Erick Fre			85.0
17130800	Félix			89.0
17130433	Hernán			86.0
17130852	Olivia			82.0

Nuevo

Matricula:

Nombre:


Apellidos:

Edad:

Promedio:

Guardar

Cancelar



Estatus de la conexión:Conectado

5 registros

Al dar click al botón Nuevo se desplegará el menú emergente el cual sirve para dar de alta un Alumno nuevo, esto de igual manera se puede realizar tanto en Materias como en Kardex respectivamente,

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Materias

ID Materia	Materia	Creditos
ED1	Estructura de datos	8
IA1	Inteligencia Artificial	8
TBD1	Taller de Bases de datos	9
TAP1		

Nuevo

ID Materia:

Materia:

Creditos:

Guardar Cancelar

Estatus de la conexión: Conectado

4 registros

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Kardex

ID Calific	Matricula	Nombre	Apellidos	ID Materia	Materia	Periodo	Calificación
8	17130433	Hernán	Hernandez	IA1	Inteligencia Artificial	2020A	91
7	17130433	Hernán	Hernandez	TAP1	Tópicos Avanzados De Progr...	2020A	95
6	17130433					2020A	81
5	17130433					2020A	78
12	17130800					2020A	90
11	17130800					2020A	94
10	17130800					2020A	86
9	17130800					2020A	88
3	17130852					2020A	81
2	17130852					2020A	75
1	17130852					2020A	91
16	17130853					2020B	100
15	17130853					2020B	70
13	18171548					2020A	74

Nuevo

Alumno: --Seleccione--

Materia: --Seleccione--

Periodo: 2018A

Calificación:

Guardar Cancelar

Estatus de la conexión: Conectado

14 registros

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Alumnos

Matricula	Nombre	Apellidos	Edad	Promedio
18171548	Chonchito	Renigris	26	74.0
17130853	Erick Francisco	Villegas Pedroza	22	85.0
17130800	Félix	Martínez Hinojo	22	89.0
17130433	Hernán	Hernández	21	86.0
17130852	Olivia	Oliveras Oliveira	19	82.0

Editar

Matricula: 18171548

Nombre: Chonchito

Apellidos: Renigris

Edad: 26


Promedio: 74.0


Guardar Cancelar


Estatus de la conexión: Conectado

5 registros

Al hacer click al botón Editar podemos modificar y/o corregir los atributos de cada caso ya sea del Alumno, las Materias o el Kardex.

**Alumnos**

**Materias**

**Kardex**

Materias


ID Materia	Materia	Creditos
ED 1	Estructura de datos	8
IA 1	Inteligencia Artificial	8
TBD 1	Taller de Bases de datos	9
TAP 1	Tópicos Avanzados De Programación	6

Editar

ID Materia: TBD 1

Materia: Taller de Bases de datos

Creditos: 9



Guardar

Cancelar

Estatus de la conexión:Conectado

4 registros

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Kardex

Alumnos

Materias

Kardex

Editar

Alumno: Chonchito Renigris

Materia: Estructura de datos

Periodo: 2020A

Calificación: 74

Guardar Cancelar

ID Calific	Matricula	Nombre	Apellidos	Edad	Periodo	Calificación
8	17130433				2020A	91
7	17130433				2020A	95
6	17130433				2020A	81
5	17130433				2020A	78
12	17130800				2020A	90
11	17130800				2020A	94
10	17130800				2020A	86
9	17130800				2020A	88
3	17130852				2020A	81
2	17130852				2020A	75
1	17130852				2020A	91
16	17130853				2020B	100
15	17130853	Erick Francisco	Villegas Pedroza	TAP 1	Tópicos Avanzados De Progr...	70
13	18171548	Chonchito	Renigris	ED 1	Estructura de datos	74

Estatus de la conexión: Conectado

14 registros

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Alumnos

Matricula	Nombre	Apellidos	Edad	Promedio
18171548	Chonchito	Renigris	26	74.0
17130853	Erick Francisco	Villegas Pedroza	22	85.0
17130800	Félix	Martínez Hinojo	22	89.0
17130433	Hernán	Hernández	21	86.0
17130852	Olivia	Olivares Oliveira	19	82.0

Eliminar

Eliminar el registro seleccionado?

Sí No

Estatus de la conexión: Conectado

5 registros

El botón Eliminar tiene la función de borrar el elemento seleccionado por el usuario.

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Materias

ID Materia	Materia	Creditos
ED1	Estructura de datos	8
IA1	Inteligencia Artificial	8
TBD1	Taller de Bases de datos	9
TAP1	Tópicos Avanzados De Programación	6

Eliminar

Eliminar el registro seleccionado?

Estatus de la conexión: Conectado

4 registros

Sistema De Calificaciones

Archivo Herramientas Ayuda

Nuevo Editar Eliminar Inicio Filtrar

Kardex

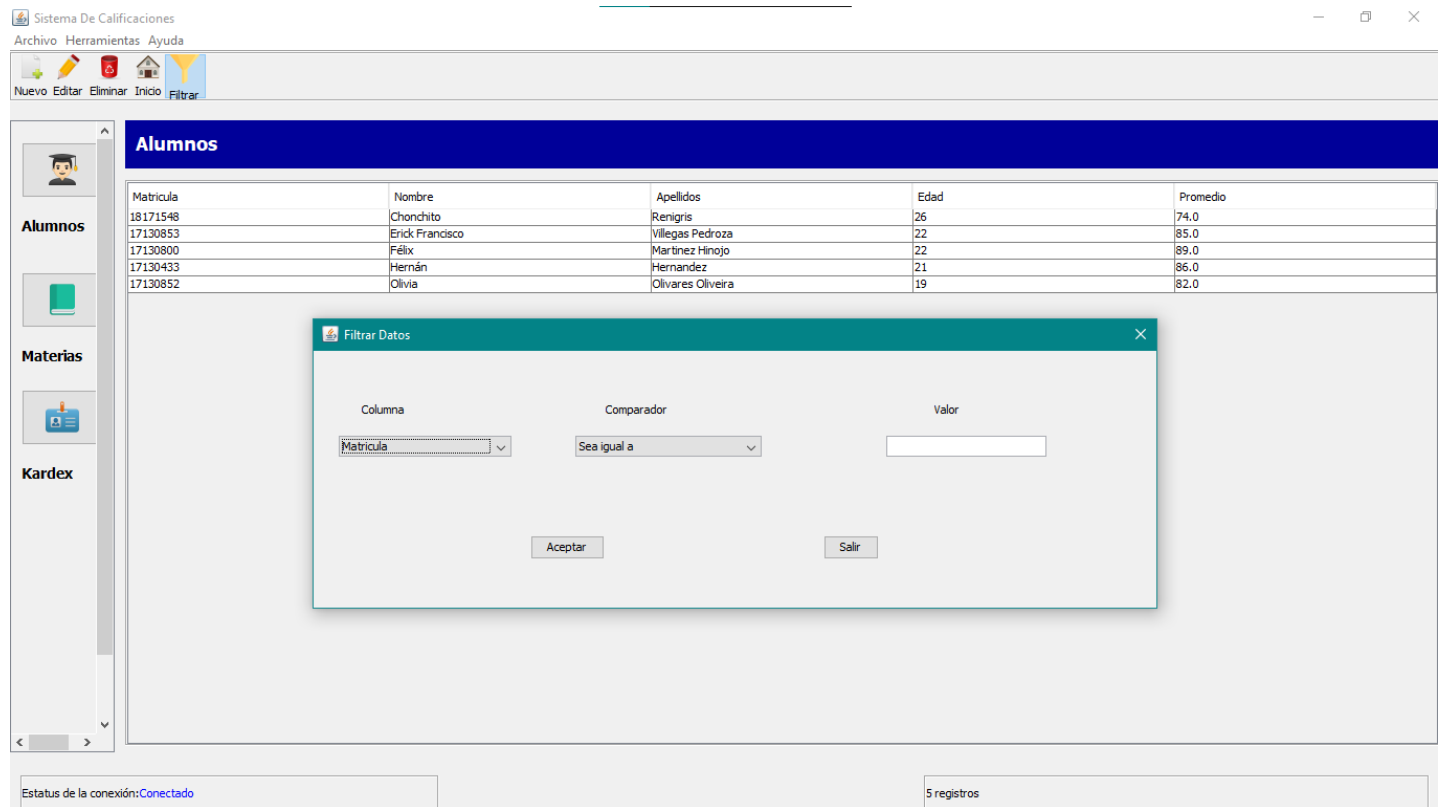
ID Calific	Matricula	Nombre	Apellidos	ID Materia	Materia	Periodo	Calificación
8	17130433	Hernán	Hernandez	IA1	Inteligencia Artificial	2020A	91
7	17130433	Hernán	Hernandez	TAP1	Tópicos Avanzados De Progr...	2020A	95
6	17130433	Hernán	Hernandez	ED1	Estructura de datos	2020A	81
5	17130433	Hernán	Hernandez	TBD1	Taller de Bases de datos	2020A	78
12	17130800	Félix	Martinez Hinojo	IA1	Inteligencia Artificial	2020A	90
11	17130800	Félix	Martinez Hinojo	TAP1	Tópicos Avanzados De Progr...	2020A	94
10	17130800	Félix	Martinez Hinojo	ED1	Estructura de datos	2020A	86
9	17130800	Félix	Martinez Hinojo	TBD1	Taller de Bases de datos	2020A	88
3	17130852	Olivia			Tópicos Avanzados De Progr...	2020A	81
2	17130852	Olivia			Estructura de datos	2020A	75
1	17130852	Olivia			Taller de Bases de datos	2020A	91
16	17130853	Erick Francisco			Estructura de datos	2020B	100
15	17130853	Erick Francisco			Tópicos Avanzados De Progr...	2020B	70
13	18171548	Chonchito			Estructura de datos	2020A	74

Eliminar

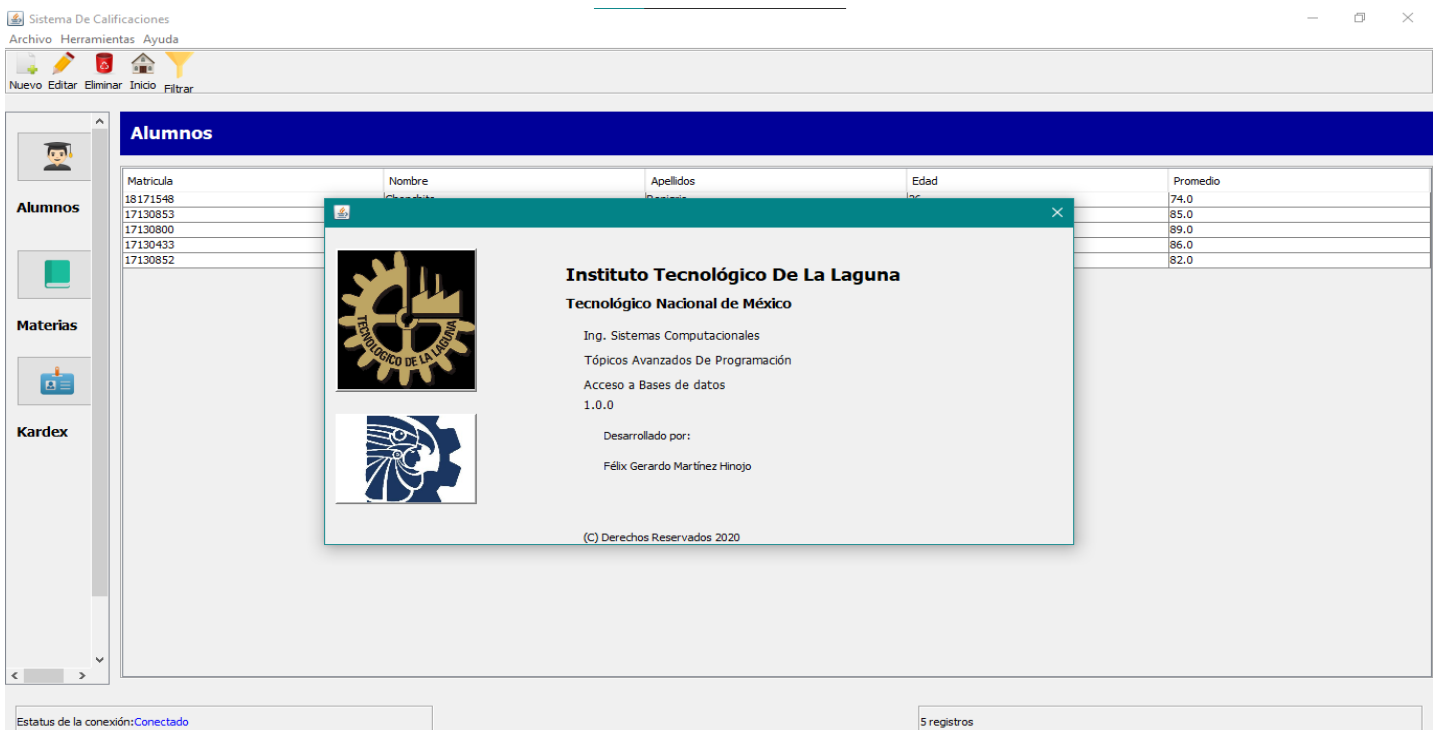
Eliminar el registro seleccionado?

Estatus de la conexión: Conectado

14 registros



El botón Filtrar es el encargado de mostrar la ventana emergente Filtrar, la cual proporciona al usuario la posibilidad de hacer consultas.



Ventana Emergente Acerca De..., el cual proporciona la visualización de los datos del desarrollador.

Anexos

script-Poblar las tablas en bd TAPdb.sql

```

/*-----
:*
:*          TECNOLOGICO NACIONAL DE MEXICO
:*          INSTITUTO TECNOLOGICO DE LA LAGUNA
:*          INGENIERIA EN SISTEMAS COMPUTACIONALES
:*          TOPICOS AVANZADOS DE PROGRAMACION "B"
:*
:*          SEMESTRE: AGO-DIC/2020      HORA: 17-18 HRS
:*
:*          Script de Población de Tablas de la BD TAPdb
:*
:* Archivo      : script-Poblar las tablas en bd TAPdb.sql
:* Autor        : Félix Gerardo Martínez Hinojo      17130800
:* Fecha        : 03/Dic/2020
:* Compilador   :
:* Descripción  : Script SQL para poblar las tablas Alumnos, Materias y Kardex
:* Última modif:
:* Fecha        Modificó          Motivo
:*=====
:* 03/Dic/2020 Félix Martínez      Se agrego el prologo
:*-----*/

/* elimina todos los registros de las tablas. */
DELETE FROM Kardex;
DELETE FROM Alumnos;
DELETE FROM Materias;

/* Poblar la tabla con registros de prueba. */

INSERT INTO Alumnos VALUES ( '17130852', 'Olivia', 'Olivares Oliveira', 19, 79.0 );
INSERT INTO Alumnos VALUES ( '17130433', 'Hernán', 'Hernandez', 21, 82.54 );
INSERT INTO Alumnos VALUES ( '17130800', 'Félix', 'Martínez H.', 22, 90.0 );

INSERT INTO Materias VALUES ( 'ED1', 'Estructura de datos', 8 );
INSERT INTO Materias VALUES ( 'TAP1', 'Tópicos avanzados de programación', 6 );
INSERT INTO Materias VALUES ( 'TBD1', 'Taller de B. de datos', 10 );
INSERT INTO Materias VALUES ( 'IA1', 'Inteligencia Artificial', 8 );

INSERT INTO Kardex VALUES ( 1, '17130852', 'TBD1', '2020A', 91 );
INSERT INTO Kardex VALUES ( 2, '17130852', 'ED1', '2020A', 75 );
INSERT INTO Kardex VALUES ( 3, '17130852', 'TAP1', '2020A', 81 );
INSERT INTO Kardex VALUES ( 4, '17130852', 'IA1', '2020A', 78 );

INSERT INTO Kardex VALUES ( 5, '17130433', 'TBD1', '2020A', 78 );
INSERT INTO Kardex VALUES ( 6, '17130433', 'ED1', '2020A', 81 );
INSERT INTO Kardex VALUES ( 7, '17130433', 'TAP1', '2020A', 75 );
INSERT INTO Kardex VALUES ( 8, '17130433', 'IA1', '2020A', 91 );

INSERT INTO Kardex VALUES ( 9, '17130800', 'TBD1', '2020A', 88 );
INSERT INTO Kardex VALUES ( 10, '17130800', 'ED1', '2020A', 86 );
INSERT INTO Kardex VALUES ( 11, '17130800', 'TAP1', '2020A', 94 );
INSERT INTO Kardex VALUES ( 12, '17130800', 'IA1', '2020A', 73 );

```