

EJERCICIO RESUELTO

Módulo: Programación

Gestión BD de Alumnos

Descripción:

En la presente unidad hemos visto como conectarnos con distintos SGBD como puede ser MySQL y como realizar sobre el operaciones de DML.

Vamos a crear una aplicación que nos permita gestionar una base de datos de Alumnos de manera funcional.

Por lo tanto en este ejercicio vamos a realizar las siguientes tareas:

- Crear una BD en MySQL de Alumnos.
- Realizar una clase que conecte con el SGBD MySQL.
- Realizar una clase que construya un menú con las siguientes opciones:
 - Mostrar todos los Alumnos
 - Crear un nuevo alumno
 - Actualizar la información de un alumno
 - Eliminar un alumno
 - Buscar un alumno por su tutor y/o apellidos.
- Para cada alumno necesitamos los siguientes datos:
 - ``id_alumno` int(5)`
 - ``nombre` varchar(20)`
 - ``apellidos` varchar(50)`
 - ``media_notas` double,`
 - ``tutor` varchar(50)`
- Usaremos un entorno gráfico (a través de la librería SWING) para programar toda la funcionalidad requerida.

Objetivos:

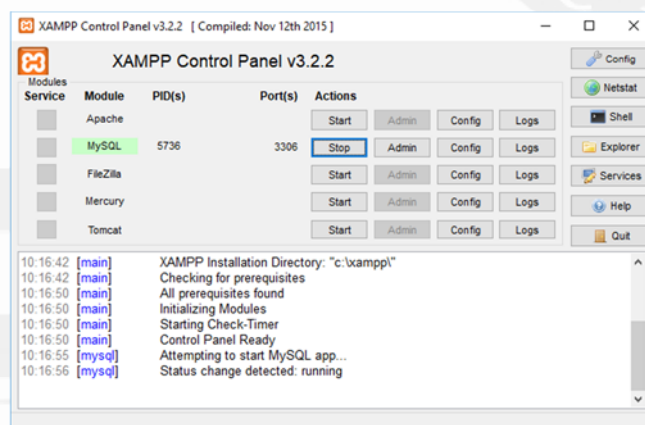
- Usar sentencias SQL para el DML (Data Manipulation Language).
- Establecer la conexión.
- Realizar una aplicación completa para la gestión de una BD.

Recursos:

- Acceso a Internet.
- Netbeans.
- JDBC.
- Driver MySQL.

Resolución:

Lo primero que debemos realizar es levantar el un servicio SQL, el de XAMPP, por ejemplo.



Después creamos la base de datos y la tabla:

```

MariaDB [(none)]> create database alumnos;
Query OK, 1 row affected (0.00 sec)
  
```

```

MariaDB [(none)]> use alumnos;
Database changed
  
```

```

MariaDB [alumnos]> create table alumno
-> (
-> id_alumno int(5) primary key,
-> nombre varchar(20),
-> apellidos varchar(50),
-> media_notas double(4,2),
-> tutor varchar(50)
-> );
Query OK, 0 rows affected (0.14 sec)
  
```

Ahora, creamos un proyecto Java y añadimos en el la librería de cliente de mariadb. Creamos una clase "Database" para la interacción con la base de datos, que sería así:

```

package gestion_de_alumnos;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;

public class Database {
    private Connection conn;

    public Database() {
        try {
            Class.forName("org.mariadb.jdbc.Driver");

            conn = DriverManager.getConnection(
                "jdbc:mariadb://localhost/alumnos",
                "root", ""
            );
        } catch (ClassNotFoundException | SQLException e) {
            System.exit(0);
        }
    }

    public String showAll() {
        String result = "-----Mostrando todos"
            + " los vehiculos-----\n";
        String q = "SELECT * from alumno";
        return result+executeSelect(q);
    }

    public String addStudent(int id, String nombre,
        String apellidos, double media, String tutor) {
        String result = "";
        result += "-----Añadiendo el alumno "
            + "con id "+id+"-----\n";
        String query = "INSERT INTO alumno values ("
            +id+", '"+nombre+"', '"+apellidos+"', '"
            +media+"', '"+tutor+"'";
        return result+executeQuery(query, "insertado");
    }

    public String updateStudent(int id, String nombre,
        String apellidos, double media, String tutor) {
        String result = "-----Modificando el alumno"
            + "con id "+id+"-----\n";
        String query = "update alumno set "
            + "nombre='"+nombre+"', "+ "apellidos='"+apellidos+"',
            + "media_notas='"+media+"', tutor='"+tutor+"'"
            + "where id_alumno='"+id+"'; ";
        return result+executeQuery(query, "actualizado");
    }
}

```

```

public String updateStudent(int id, String nombre,
    String apellidos, double media, String tutor){
    String result = "-----Modificando el alumno"
        + "con id "+id+"-----\n";
    String query = "update alumno set "
        + "nombre='"+nombre+"', "+ "apellidos='"+apellidos+"',
        + "media_notas='"+media+"', tutor='"+tutor+"' "
        + "where id_alumno='"+id+"'; ";
    return result+=executeQuery(query, "actualizado");
}

public String deleteStudent(int id){
    String result = "-----Eliminado el alumno"
        + "con id "+id+"-----\n";
    String query = "delete from alumno "
        + "where id_alumno="+id+"; ";
    return result+=executeQuery(query, "eliminado");
}

public String searchStudent(int id){
    String result = "-----Mostrando el alumno"
        + "con id "+id+"-----\n";
    String q = "SELECT * from alumno where id_alumno="+id+";";
    return result+=executeSelect(q);
}

private String executeQuery(String q, String action){
    try{
        int n = conn.prepareStatement(q).executeUpdate();
        return "\nSe han "+action+" "+n+" alumnos";
    }catch(SQLException e){
        return "\nNo se pudo realizar la operacion";
    }
}

private String executeSelect(String q){
    try{
        String resultTxt = "";
        ResultSet result = conn.createStatement().executeQuery(q);
        resultTxt+="\nAlumnos encontrados en la BD: \n";
        while(result.next()){
            resultTxt+="\nId: "+result.getInt(1)+
                "\n\tNombre: "+result.getString(2)+
                "\n\tApellidos: "+result.getString(3)+
                "\n\tMedia: "+result.getDouble(4)+
                "\n\tTutor: "+result.getString(5);
        }
        return resultTxt;
    }catch(SQLException e){
        return "No se pudo realizar la consulta";
    }
}
}

```

Después, creamos la interfaz con swing, que tendrá este aspecto:

La clase, quedará así:

```
package gestion_de_alumnos;

public class GUI extends javax.swing.JFrame {
    Database db;

    public GUI() {
        db = new Database();
        initComponents();
    }
    @SuppressWarnings("unchecked")
    Generated Code

    private void addStudent_buttonActionPerformed(java.awt.event.ActionEvent evt) {
        int id = Integer.parseInt(this.studentId_text.getText());
        String name = this.studentName_text.getText();
        String surname = this.studentSurname_text.getText();
        double average = Double.valueOf(this.studentAverage_text.getText());
        String tutor = this.studentTutor_text.getText();
        String txt = this.db.addStudent(id, name, surname, average, tutor);
        this.result_textArea.setText(txt);
    }

    private void showAllButtonActionPerformed(java.awt.event.ActionEvent evt) {
        String txt = this.db.showAll();
        this.result_textArea.setText(txt);
    }
}
```

```

private void updateStudent_buttonActionPerformed(java.awt.event.ActionEvent evt) {
    int id = Integer.parseInt(this.studentId_text.getText());
    String name = this.studentName_text.getText();
    String surname = this.studentSurname_text.getText();
    double average = Double.valueOf(this.studentAverage_text.getText());
    String tutor = this.studentTutor_text.getText();
    String txt = this.db.updateStudent(id, name, surname, average, tutor);
    this.result_textArea.setText(txt);
}

private void deleteStudent_buttonActionPerformed(java.awt.event.ActionEvent evt) {
    int id = Integer.parseInt(this.studentIdToFind_text.getText());
    String txt = this.db.deleteStudent(id);
    this.result_textArea.setText(txt);
}

private void searchStudent_buttonActionPerformed(java.awt.event.ActionEvent evt) {
    int id = Integer.parseInt(this.studentIdToFind_text.getText());
    String txt = this.db.searchStudent(id);
    this.result_textArea.setText(txt);
}

public static void main(String args[]) { ...32 lines }

// Variables declaration - do not modify
private javax.swing.JButton addStudent_button;
private javax.swing.JButton deleteStudent_button;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTextArea result_textArea;
private javax.swing.JButton searchStudent_button;
private javax.swing.JButton showAllButton;
private javax.swing.JLabel studentAverage_label;
private javax.swing.JTextField studentAverage_text;
private javax.swing.JLabel studentIdToFind_label;
private javax.swing.JTextField studentIdToFind_text;
private javax.swing.JLabel studentId_label;
private javax.swing.JTextField studentId_text;
private javax.swing.JLabel studentName_label;
private javax.swing.JTextField studentName_text;
private javax.swing.JLabel studentSurname_label;
private javax.swing.JTextField studentSurname_text;
private javax.swing.JLabel studentTutor_label;
private javax.swing.JTextField studentTutor_text;
private javax.swing.JButton updateStudent_button;
// End of variables declaration
}

```