

TEMA 2.6

Por Eduardo Felipe Poot Chairez

Se tienen dos tablas (un maestro puede impartir una materia, pero una materia puede ser impartida por muchos maestros):

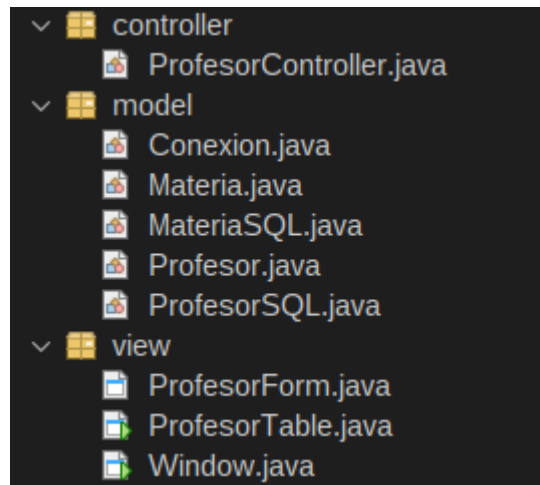
```
CREATE TABLE Materias (  
    ID_Materia VARCHAR(10) PRIMARY KEY,  
    Nombre_Materia VARCHAR(100) NOT NULL,  
    Creditos INT  
);  
  
CREATE TABLE Profesores (  
    ID_Profe VARCHAR(10) PRIMARY KEY,  
    Nombre VARCHAR(50) NOT NULL,  
    Apellido VARCHAR(50) NOT NULL,  
    Email VARCHAR(100),  
    ID_Materia VARCHAR(10),  
    FOREIGN KEY (ID_Materia) REFERENCES Materias(ID_Materia)  
);
```

Se insertan cinco materias.

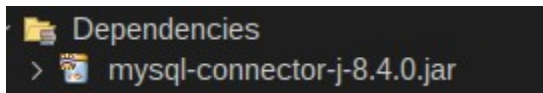
```
INSERT INTO Materias (ID_Materia, Nombre_Materia, Creditos)  
VALUES  
    ('MAT101', 'Cálculo Diferencial', 5),  
    ('PROG202', 'Programación Orientada a Objetos', 6),  
    ('BD303', 'Bases de Datos', 4),  
    ('RED404', 'Redes de Computadoras', 4),  
    ('IA505', 'Inteligencia Artificial', 6);
```

Se utilizó la estructura MVC (Model–View–Controller), en la cual se modularizó el programa de la siguiente manera:

- En la carpeta **vista** se encuentran la ventana de inicio, el formulario de inserción (INSERT) y la tabla para visualizar, eliminar y modificar los datos.
- En la carpeta **modelo** se encuentran los archivos que manejan la lógica de conexión y la recuperación de los datos de la base de datos MySQL.
- En la carpeta **controlador** se encuentra el archivo que maneja la lógica encargada de relacionar el contenido del modelo y la vista.



Para realizar la conexión con MySQL, se utilizó la dependencia correspondiente y el archivo **Conexion.java**.



```
1 package model;
2 import java.sql.Connection;
3 import java.sql.DriverManager;
4
5 public class Conexion {
6
7     private static final String URL =
8         "jdbc:mysql://localhost:3306/tema2_6?useSSL=false&serverTimezone=UTC";
9     private static final String USER = "eduardo";
10    private static final String PASS = "108310";
11
12    public static Connection getConexion() {
13        try {
14            return DriverManager.getConnection(URL, USER, PASS);
15        } catch (Exception e) {
16            System.out.println("Error de conexión: " + e.getMessage());
17            return null;
18        }
19    }
20 }
21
22 }
```

La interfaz principal luce de la siguiente manera.



Dentro del programa se pueden realizar todas las funciones del CRUD, tales como:

- **Insertar**

The "Registrar Profesor" dialog box is shown. It has a title bar with the text "Registrar Profesor". Inside, the title "Nuevo Profesor" is displayed. The form includes the following fields:

- ID: 11
- Nombre: Mario
- Apellido: Moreno
- Email: mario@gmail.com
- Materia: Cálculo Diferencial (selected from a dropdown menu)

At the bottom, there are two buttons: "Guardar" and "Cancelar".

- **Visualizar** (donde se relacionan las dos tablas mediante el uso de una clave foránea)

The "Lista de Profesores" window displays a table titled "Lista de Profesores y Materias". The table has the following columns: ID Profesor, Nombre, Apellido, Email, ID Materia, Materia, Modificar, and Eliminar. The data shown is as follows:

ID Profesor	Nombre	Apellido	Email	ID Materia	Materia	Modificar	Eliminar
11	Mario	Moreno	mario@gmail.com	MAT101	Cálculo Diferencial	Modificar	Eliminar

Below the table is a large empty rectangular area. At the bottom right of the window, there is a "Cerrar" button.

- **Modificar**

Modificar Profesor

ID: 11

Nombre: Mario

Apellido: Moreno

Email: mario@gmail.com

Materia: Cálculo Diferencial

Actualizar Cancelar

- **Eliminar** (Se presiona el botón de la esquina y se despliega una advertencia para confirmar o denegar la eliminación.)

Clave	Nombre	Apellido	Email	ID Materia	Materia	Modificar	Eliminar
11	Mario	Moreno	mario@gmail.com	MAT101	Cálculo Diferencial	Modificar	Eliminar

Confirmar eliminación

¿Eliminar el profesor con clave 11?

Yes No