

Ejercicio 1 - Estudiantes

Queremos conectarnos a la base de datos de un instituto llamada **school**, que está compuesta de una tabla **student**, que contiene la información de los diferentes estudiantes.

La tabla puede crearse con las siguientes sentencias:

```
DROP DATABASE IF EXISTS school;
```

```
CREATE DATABASE school;
```

```
USE school;
```

```
CREATE TABLE student (  
    id CHAR(9) PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    surname VARCHAR(100) NOT NULL,  
    age INT NOT NULL  
);
```

Los datos de los estudiantes serían los siguientes:

```
INSERT INTO student VALUES ('11111111A', 'Draco', 'Malfoy', 25);
```

```
INSERT INTO student VALUES ('22222222B', 'Hermione', 'Granger', 23);
```

```
INSERT INTO student VALUES ('33333333C', 'Harry', 'Potter', 20);
```

```
INSERT INTO student VALUES ('44444444D', 'Ron', 'Weasley', 22);
```

Debes diseñar una aplicación en Java llamada **app-students** que se conecte a la base de datos y permita realizar varias operaciones sobre ella. Para ello, debes definir una clase **Student** que disponga de los siguientes atributos:

- **id**: String. → hace referencia al DNI.
- **name**: String.
- **surname**: String.
- **age**: int.

También debes añadirle los métodos que consideres necesarios.

Por otra parte, deberás crear una clase llamada **ManageStudents**, que permitirá realizar diferentes operaciones sobre la base de datos del instituto. La clase tendrá, como mínimo, los siguientes atributos y métodos:

-Connection connection.

+void openConnection().

+void closeConnection().

+boolean addStudent(Student student): añade un nuevo estudiante a la base de datos.

+Student getStudent(String id): obtiene la información de un estudiante determinado.

+boolean deleteStudent(String id): borra la información de un estudiante determinado.

+boolean modifyStudent(Student student): modifica los datos de un estudiante determinado, si ya existe en la base de datos.

+ArrayList<Student> getStudentsList(): devuelve una lista con todos los estudiantes almacenados en la BD.

Después, debes definir una clase **AppStudents** en la que le muestres al usuario un menú con las siguientes opciones:

1.- MATRICULAR UN ESTUDIANTE.

2.- DAR DE BAJA A UN ESTUDIANTE.

3.- ACTUALIZAR DATOS DE UN ESTUDIANTE.

4.- VER DATOS DE UN ESTUDIANTE.

5.- VER DATOS DE TODOS LOS ESTUDIANTES.

6.- SALIR.

En el caso de que no sea posible realizar alguna de las operaciones, debemos mostrarle un programa descriptivo al usuario y continuar con la ejecución de la aplicación.