# CASO PRÁCTICO 1

**BASE DE DATOS DE ALUMNOS**

## Contexto

En una academia necesitan una aplicación que almacene los datos sobre sus alumnos. Dicha aplicación deberá poder realizar una serie de consultas sobre estos datos de alumnos.

## Cuestiones a resolver

1. Crear la siguiente base de datos Alumnos del esquema aad en MySQL (consultar **anexo 1**):

[Anexo 1](https://cesurformacion0.sharepoint.com/:b:/r/sites/Contenidos399/Documentos%20compartidos/General/2_Contenidos%20FP/4_REVISI%C3%93N%20CONTENIDOS%20FP%202021-XXXX/2024/T.S.%20DAM%202%C2%BA/ACCESO%20A%20DATOS/MATERIAL%20ACTUALIZADO/UD2_AD_20240906/Evaluaci%C3%B3n/Material%20Caso1_UD2.pdf?csf=1&web=1&e=P9yJyv)



Introduce algunos datos, siendo el primero tu nombre con los apellidos y la edad.

1. Descargar y agregar el conector para MySQL.
2. Crear un proyecto java llamado Acceso\_BBDD\_MySQL con dos clases una principal y otra llamada Acceso. Añadir el conector al proyecto.
3. La clase acceso es la que se conecta con el servidor MySQL, esquema aad y tabla alumnos:

**import** java**.**sql**.\*;**

public class Acceso**{**

private Connection conexion**;**

private Statement St**;**

private ResultSet rs**;**

public Acceso**(){**

**}**

public void Conectar**(){**

**try{**

// Se carga el driver mysql de la siguiente forma:

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

// Datos para la conexión:

String url**=**"jdbc:mysql://localhost/aad"**;**

String usuario**=**"root"**;**

String contraseña**=**"root"**;**

// Crear un objeto conexión:

conexion **=** DriverManager**.**getConnection **(**url**,**usuario**,** contraseña**);**

// Se crea un objeto de tipo Statement, para realizar la consulta:

St **=** conexion**.**createStatement**();**

**}catch** **(**Exception e**){**

e**.**printStackTrace**();**

**}**

**}**

public void RealizarConsulta**(**String consulta**){**

**try{**

// Se realiza la consulta. Los resultados se guardan en el ResultSet rs

rs **=** St**.**executeQuery **(**consulta**);**

// se recuperan los datos recorriendo el ResultSet, mostrando por pantalla los resultados:

**while** **(**rs**.**next**()){**

System**.**out**.**println **(**rs**.**getInt **(**1**)** **+** " " **+**

rs**.**getString **(**2**)** **+** " " **+**

rs**.**getString**(**3**)** **+** " " **+**

rs**.**getInt **(**4**)** **);**

**}**

// Se debe cerrar la conexión con la base de datos (aquí o en otra función aparte, al finalizar todas las operaciones)

//St.close();

//rs.close();

//conexion.close();

**}catch** **(**Exception e **)** **{**

e**.**printStackTrace**();**

**}**

**}**

**}**

Usar esta clase desde la función main para mostrar los datos de la tabla Alumnos mediante una consulta SQL. El resultado debe ser similar al siguiente:



1. Explicar detalladamente qué hace el bucle while de RealizarConsulta.
2. Modificar el código de la clase Acceso para que muestre el número de columnas, nombre de la tabla y esquema o catálogo, así como el nombre de las columnas de la tabla tal y como se muestra.



**Nota:** será necesario crear un objeto de tipo ResultSetMetaData

**Ejemplo:** ResultSetMetaData rsMetaData = rs.getMetaData();

Se deja como opcional investigar la forma de mostrar el número de filas. No te olvides de agregar el cierre de la conexión cuando termines las consultas.

## Recursos

Se deberá consultar el contenido de la unidad, internet, libros, revistas y utilizar medios informáticos para la presentación del caso práctico (Word, Power-Point…).

## Objetivos

Desarrollar la capacidad de conectar una aplicación Java con una base de datos MySQL, realizar consultas SQL y manejar los resultados y metadatos de manera eficiente.

## Resultados de aprendizaje y criterios de evaluación

Desarrolla aplicaciones que gestionan información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión:

* Se ha utilizado el conector idóneo en la aplicación.
* Se ha establecido la conexión.
* Se ha definido la estructura de la base de datos.
* Se han definido los objetos destinados a almacenar el resultado de las consultas.
* Se han desarrollado aplicaciones que efectúan consultas.
* Se han eliminado los objetos una vez finalizada su función.

## Rúbrica para su evaluación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Excelente** | **Satisfactorio** | **Mejorable** | **Insuficiente** |
| **Conexión y consultas a la base de datos MySQL.** | La aplicación conecta correctamente con la base de datos MySQL y realiza las consultas solicitadas sin errores. Los datos de los alumnos se muestran de forma precisa y clara. El cierre de la conexión se realiza adecuadamente. | La aplicación conecta correctamente y realiza consultas, pero puede tener pequeños errores menores o la conexión no se cierra correctamente. | La conexión con la base de datos es inestable, o las consultas presentan errores. Los datos se muestran de manera incompleta. | No se logra conectar correctamente con la base de datos, o las consultas no producen los resultados esperados. |
| 40% | 4 | 3 | 2 | 0.25 |
| **Manejo de metadatos de la base de datos y ResultSetMetaData.** | La modificación de la clase permite mostrar correctamente el número de columnas, el nombre de la tabla, el esquema, y el nombre de las columnas, utilizando ResultSetMetaData. El código es claro y eficiente. | La modificación permite mostrar la mayoría de los metadatos solicitados, aunque puede faltar algún detalle menor o el código es poco claro. | La clase Acceso muestra información limitada de los metadatos, o no presenta todos los elementos requeridos de manera clara. | No se implementa correctamente el uso de ResultSetMetaData, y/o no se logran mostrar los metadatos requeridos. |
| 40% | 4 | 3 | 2 | 0.25 |
| **Presentación, extensión, estructura y precisión ortográfica.** | Responde de forma ordenada, comprensible y coherente, profundizando en todos los conceptos e ideas que expresa con rigor profesional y desde un punto de vista crítico y reflexivo. No comete errores gramaticales, de ortografía o puntuación. | Responde de forma ordenada, profundizando en algunos conceptos e ideas que expresa con cierta visión crítica y rigor profesional. Comete algún error gramatical, ortográfico y/o de puntuación. | Usa un lenguaje comprensible, pero desordenado y sin profundizar en las ideas. Falta rigor profesional y visión crítica del contexto. Comete varios errores ortográficos, de gramática y/o de puntuación. | Presenta una respuesta desordenada, pobre e incompleta, que dificulta la comprensión de las ideas que se exponen. Comete errores gramaticales, de ortografía y puntuación. |
| 10% | 1 | 0.75 | 0.5 | 0.25 |
| **Uso de recursos adicionales y creatividad en la respuesta.** | Utiliza numerosas fuentes de información relevantes, fiables y actualizadas. Aporta recursos adicionales que clarifican la respuesta. Responde con gran originalidad, haciendo uso de diferentes herramientas y aplicaciones, y expresando ideas ingeniosas. | Utiliza diversas fuentes de información actualizadas. Aporta algún recurso adicional. Muestra cierta originalidad en su respuesta, incluyendo nuevas ideas y puntos de vista. | Utiliza alguna fuente de información externa. Aporta alguna imagen o gráfico adicional. Se hace uso de ideas de otros autores, pero no se aportan ideas y puntos de vista propios. | No hace uso de fuentes fiables ni añade recursos adicionales en la respuesta. Se utilizan ideas de otros autores sin citar. |
| 10% | 1 | 0.75 | 0.5 | 0.25 |