Implementing an API in Express.js using MySQL (Step by Step)

Written by Mariano Perdices – 1008005

1. Setting Up the Database:

Create three tables:

Staff Table: Stores staff details (Sid [Primary Key], Name).

Dept Table: Stores department details (Code [Primary Key]).

Work Table: Links staff with departments using foreign keys (Sid, Code).

2. Establish the MySQL Connection:

In the Express.js app (you may choose to split the dept and staff like in ce2q1), establish a connection to the MySQL database.

3. Create the API Route:

Define the route in Express.js that listens for GET requests at /dept/count.

Write an SQL query to join the staff, work, and dept tables and count the number of staff for each department (**We use Left Join in this case**)

4. Run the Query:

The SQL query should group by the department (Code) and count the staff (Sid).

5. Return the Results:

Return the query results as a JSON response containing the department code and the corresponding staff count.

6. Test the API:

Run your Express app and test the endpoint by visiting localhost:3000/dept/count. The response should display the count of staff per department.