Lab 3: Advanced SQL Queries

CSL303: Database Management Systems

August 19, 2025

Objective

To gain experience with advanced SQL features, including various join types, subqueries, set operations, and data manipulation language (DML). This lab uses a pre-populated SQLite database that you will create.

Setup

You must first create the SQLite database from the provided SQL script. Run the following command in your terminal from the same directory where `lab3_data.sql` is located:

```
sqlite3 lab3.db < lab3_data.sql
```

This command will execute the SQL script, creating the necessary tables (`Students`, `Faculty`, `Courses`, `Enrolled`) and populating them with data. You can now open the database to run the queries for the exercises below.

```
sqlite3 lab3.db
```

Exercises

For each question, write a single SQL query that produces the required result.

Part 1: Joins and Outer Joins

1. List the names of all courses and the name of the faculty member who teaches each course.
2. Find the names of all students who are enrolled in a course taught by 'Prof. Sharma'.
3. List all student names. If a student is enrolled in any course, also list the course name. **Include students who are not enrolled in any course.**
4. List all faculty members and the names of the courses they teach. **Include faculty who are not currently teaching any course.**

Part 2: Advanced Conditions and Functions

1. Find all students whose name contains the letter 'a'. The search should be case-insensitive.
2. Find the student ID and name for all students who do not have a discipline listed (i.e., their discipline is NULL).
3. List the names and registration dates of all students who registered in the year 2022.
4. Find the names of all students who registered in August 2022. Use the `BETWEEN` operator.

Part 3: Subqueries and Set Operations

1. Find the names of all students who have a GPA greater than the average GPA of all students. (Use a scalar subquery).
2. Find the names of all 'CSE' students who are **not** enrolled in 'Databases' (CSL303). Use the `EXCEPT` operator.
3. Find the names of all courses that have at least one student enrolled. Use a subquery with `EXISTS`.
4. Find the names of students who have the highest GPA in their respective discipline. (Use a correlated subquery).

Part 4: Data Manipulation Language (DML)

1. A new student has joined. Insert the following record into the `Students` table:
   - sid: 108
   - sname: 'Ravi'
   - discip: 'EE'
   - gpa: 8.0
   - registration_date: '2023-09-01'
2. 'Prof. Sharma' has decided to give a 10% GPA boost to all students who received an 'A' in 'Databases' (CSL303). Write an `UPDATE` statement to reflect this change.
3. The 'Linear Algebra' course (MAL251) has been cancelled. Delete all enrollment records for this course.