# SQL Final Project: Student Course Management System

## Project Overview

You are hired as a junior database developer for an EdTech company to create a Student Course Management System. You will design the schema, populate it with data, run advanced SQL queries, and host the entire project on GitHub with proper documentation.

## Database Schema

### Students

- student\_id (INT, PRIMARY KEY)

- first\_name (VARCHAR)

- last\_name (VARCHAR)

- email (VARCHAR)

- date\_of\_birth (DATE)

### Instructors

- instructor\_id (INT, PRIMARY KEY)

- first\_name (VARCHAR)

- last\_name (VARCHAR)

- email (VARCHAR)

### Courses

- course\_id (INT, PRIMARY KEY)

- course\_name (VARCHAR)

- course\_description (TEXT)

- instructor\_id (INT, FOREIGN KEY → Instructors)

### Enrollments

- enrollment\_id (INT, PRIMARY KEY)

- student\_id (INT, FOREIGN KEY → Students)

- course\_id (INT, FOREIGN KEY → Courses)

- enrollment\_date (DATE)

- grade (CHAR(1))

## Project Tasks

### Part 1: Create the Database and Tables

- Create a new SQL database named course\_management.

- Define the schema above with proper keys and constraints.

### Part 2: Insert Sample Data

- Insert at least 10 students, 3 instructors, 5 courses, and 15 enrollments with a mix of grades.

### Part 3: Write SQL Queries

- Students who enrolled in at least one course.

- Students enrolled in more than two courses.

- Courses with total enrolled students.

- Average grade per course (A=4, ..., F=0).

- Students who haven’t enrolled in any course.

- Students with their average grade across all courses.

- Instructors with the number of courses they teach.

- Students enrolled in a course taught by “John Smith”.

- Top 3 students by average grade.

- Students failing (grade = ‘F’) in more than one course.

### Part 4: Advanced SQL

- Create a VIEW named student\_course\_summary (student name, course, grade).

- Add an INDEX on Enrollments.student\_id.

- Optional: Create a trigger or stored procedure that logs new enrollments.

## README.md Guidelines

- Project Title and Description

- ERD (Entity Relationship Diagram) - optional

- Instructions to Run the SQL Code

- Explanation of the Schema

- Descriptions of Key Queries

- Sample Output Descriptions or Screenshots

- Challenges and Lessons Learned

## Submission Instructions

- Push the entire project folder to your personal GitHub account.

- Make the repository public.

- Ensure your code is clean, organized, and well-commented.

- Add a professional README.md explaining your project.

- Submit your GitHub repository link to the course instructor.