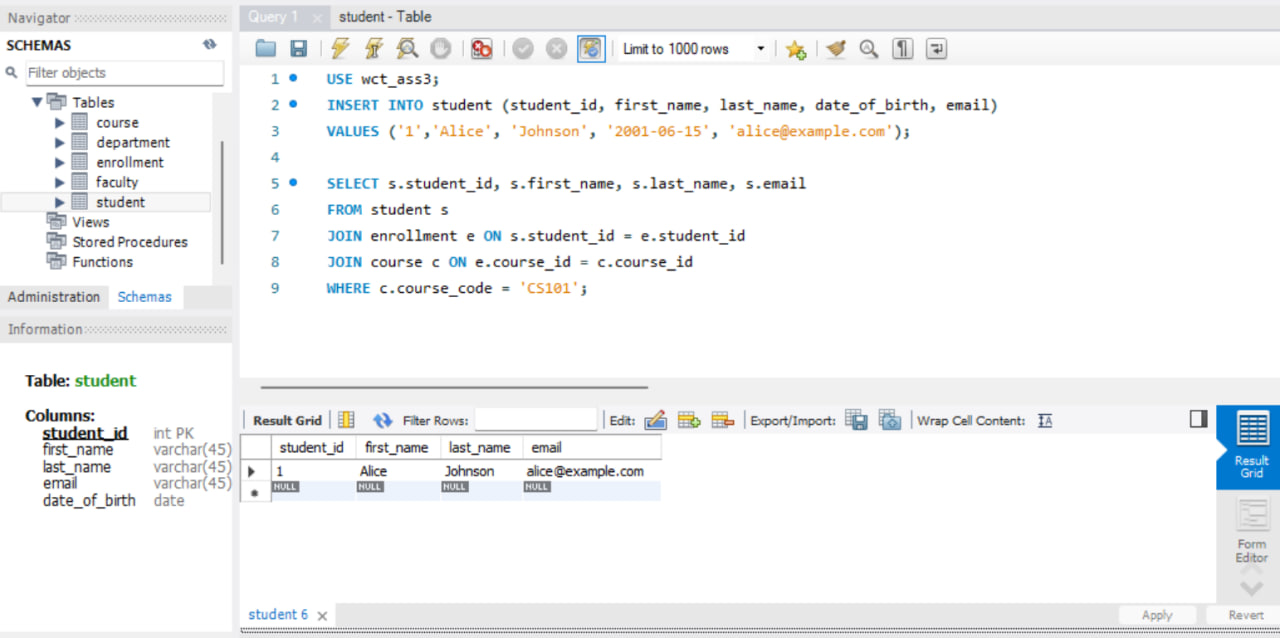
Name: Pet laiheang

Class: ITE M2

WCT II

Part 6: Querying the Database Write SQL queries to answer

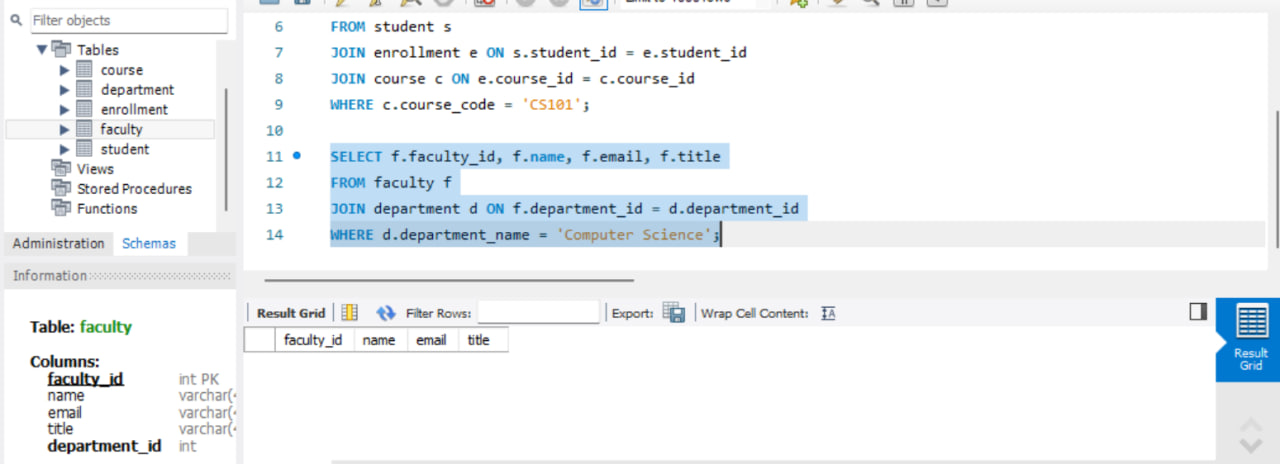
1. Retrieve all students who enrolled in a specific course.



Explanation:

* The students table contains student details.
* The enrollments table links students and courses.
* The courses table contains course details.
* We use JOIN to connect the three tables using student\_id (from students → enrollments) and course\_id (from enrollments → courses).
* WHERE c.course\_code = 'CS101' filters the results to show only students enrolled in CS101.

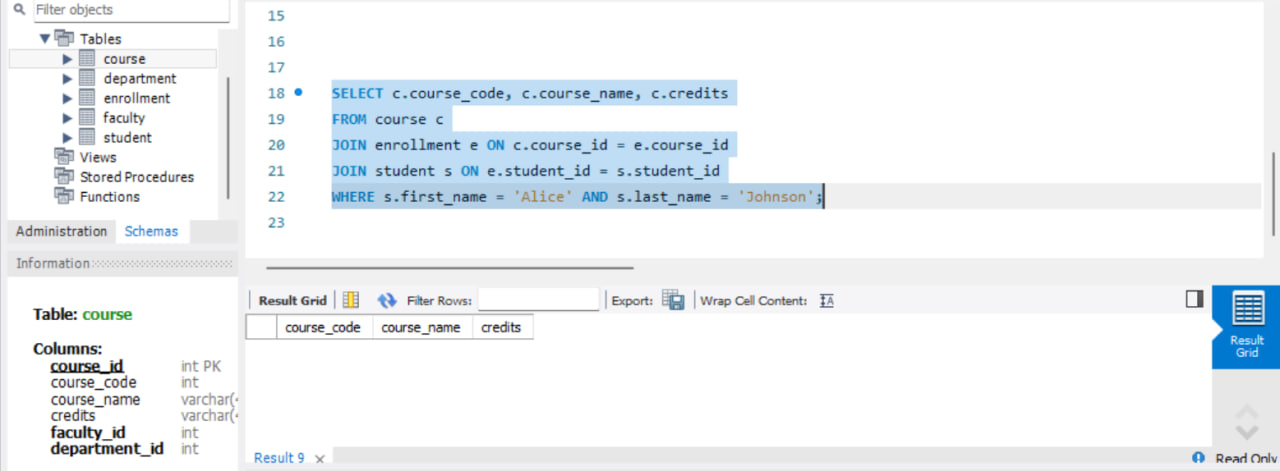
2. Find all faculty members in a particular department



Explanation:

* The faculty table contains faculty details.
* The departments table contains department details.
* We join these tables using department\_id since each faculty member belongs to one department.
* WHERE d.department\_name = 'Computer Science' filters the results to show only faculty members in the Computer Science department.

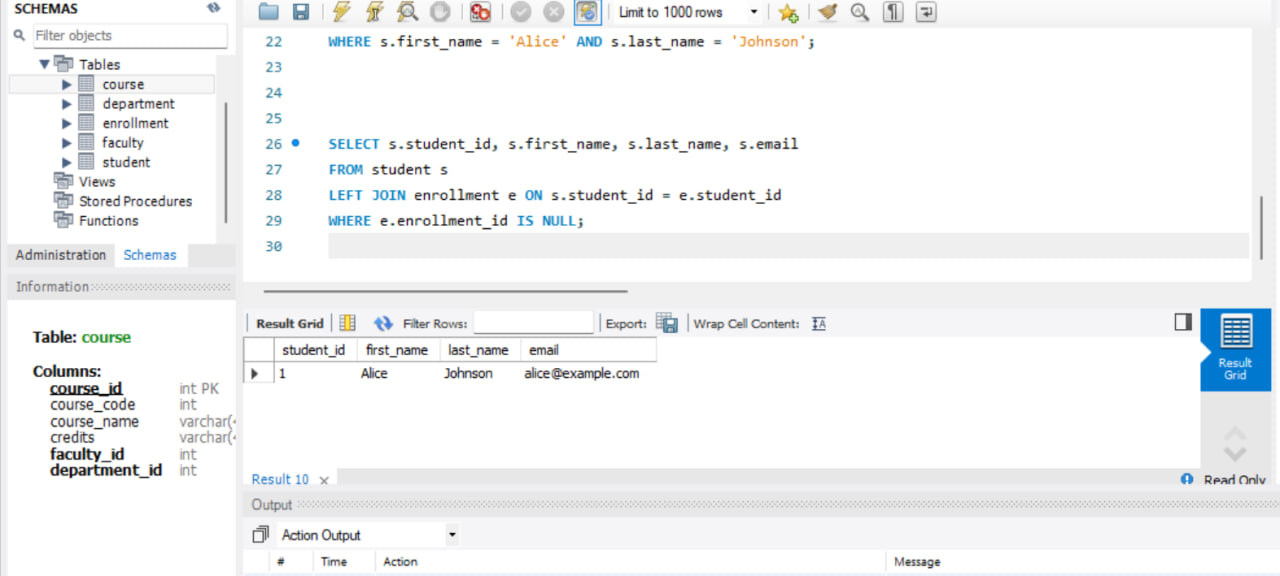
3. List all courses a particular student is enrolled in.



Explanation:

* The students table provides student names.
* The enrollments table links students to courses.
* The courses table provides course details.
* We use JOIN to connect these tables.
* WHERE s.first\_name = 'Alice' AND s.last\_name = 'Johnson' ensures we retrieve courses for Alice Johnson only.

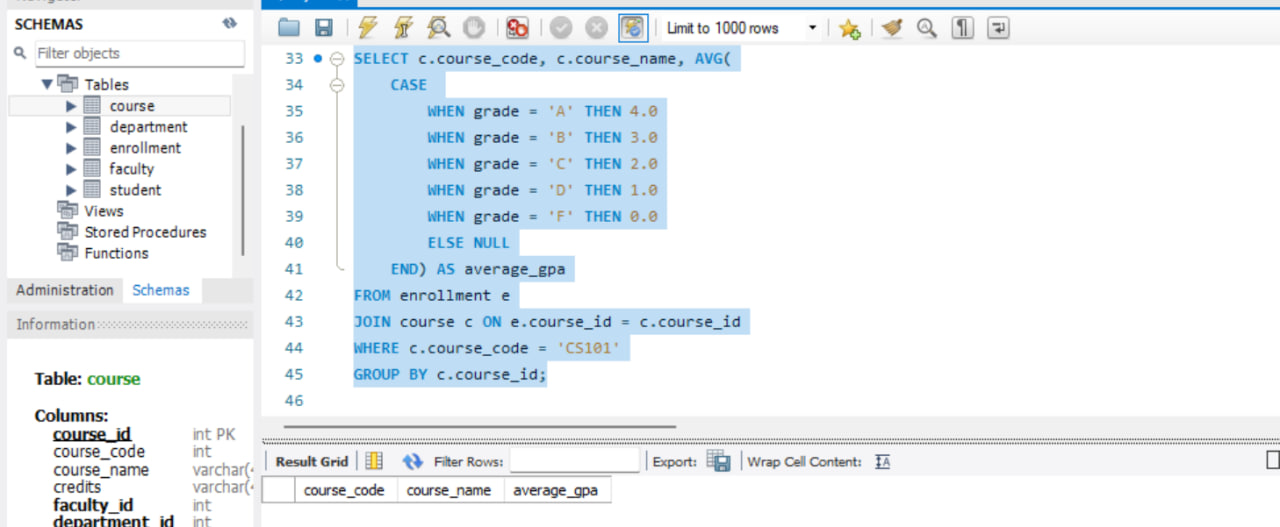
4. Retrieve students who have not enrolled in any course.



Explanation:

* We use a LEFT JOIN between students and enrollments to keep all students, even those without enrollments.
* If a student has never enrolled, their student\_id will have no matching entry in enrollments, meaning e.enrollment\_id will be NULL.
* WHERE e.enrollment\_id IS NULL filters out students who have enrolled in at least one course, showing only those who haven't.

5. Find the average grade of students in a specific course.



Explanation:

* The enrollments table contains student grades.
* The courses table contains course details.
* We use JOIN to connect courses to enrollments.
* The CASE statement converts letter grades into GPA values (A = 4.0, B = 3.0, etc.).
* AVG(...) calculates the average GPA for that course.
* WHERE c.course\_code = 'CS101' filters only for CS101.
* GROUP BY c.course\_id ensures we calculate one average per course.