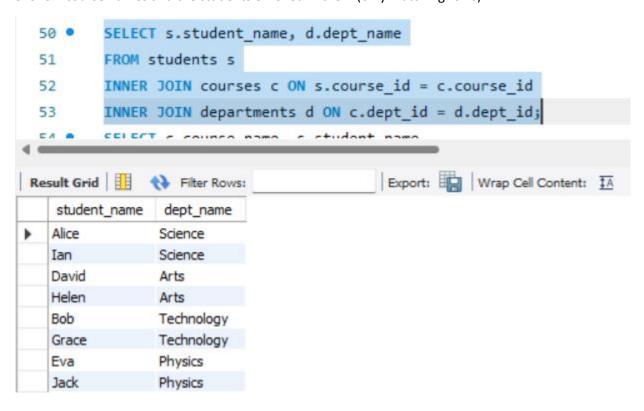
SQL JOINS 2

```
CREATE TABLE departments (
  dept id INT PRIMARY KEY,
  dept_name VARCHAR(50),
  dept head VARCHAR(50)
);
CREATE TABLE courses (
  course_id INT PRIMARY KEY,
  course_name VARCHAR(50),
  credits INT,
  dept_id INT,
  FOREIGN KEY (dept_id) REFERENCES departments(dept_id)
);
CREATE TABLE students (
  student_id INT PRIMARY KEY,
  student_name VARCHAR(50),
  course_id INT,
  FOREIGN KEY (course_id) REFERENCES courses(course_id)
);
1. Show each student's first name and their course name using an INNER JOIN.
               SELECT s.student_name, c.course_name
      45 •
               FROM students s
      46
               INNER JOIN courses c ON s.course_id = c.course_id;
      47
      48
      40
                                                        Export: Wrap Cell Content: IA
    Result Grid
                      Filter Rows:
        student_name
                        course_name
       Alice
                       Biology
                       Biology
       Ian
       David
                       History
       Bob
                       Computer Science
       Grace
                       Computer Science
                       Quantum Mechanics
       Eva
        Jack
                       Quantum Mechanics
       Helen
                       Philosophy
```

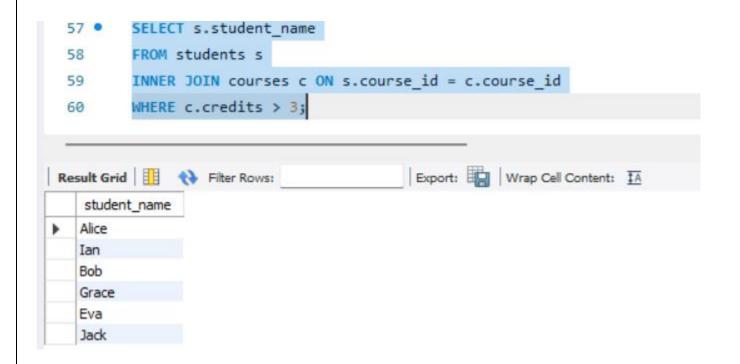
 $2.\ \mbox{Display}$ all students and the department of their course using an INNER JOIN.

```
50 •
         SELECT s.student name, d.dept name
         FROM students s
 51
         INNER JOIN courses c ON s.course_id = c.course_id
 52
         INNER JOIN departments d ON c.dept_id = d.dept_id;
         CELECT & COURSE NAME & STUDENT NAME
                                            Export: Wrap Cell Content: IA
Result Grid
               Filter Rows:
   student_name
               dept_name
  Alice
                Science
                Science
  Ian
  David
                Arts
  Helen
                Arts
  Bob
                Technology
                Technology
  Grace
  Eva
                Physics
  Jack
                Physics
```

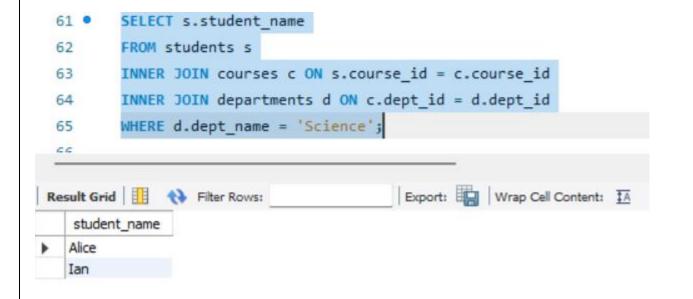
3. Show course names and the students enrolled in them (only matching rows).

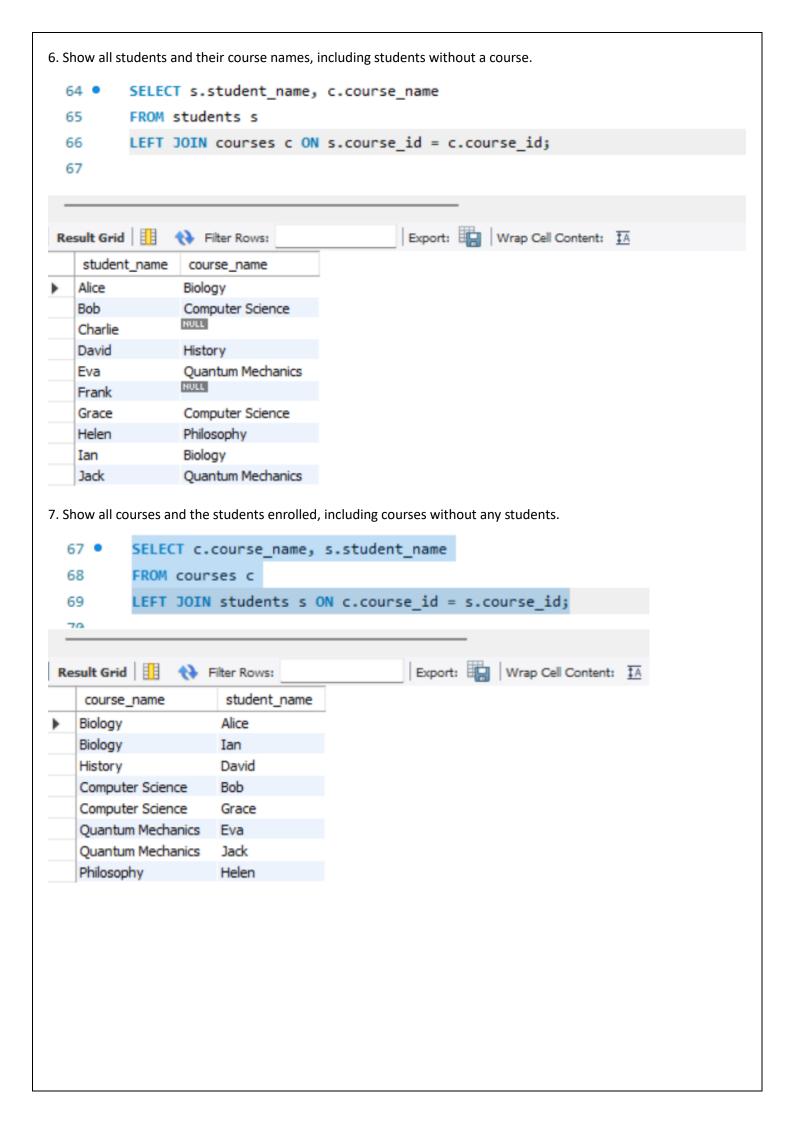


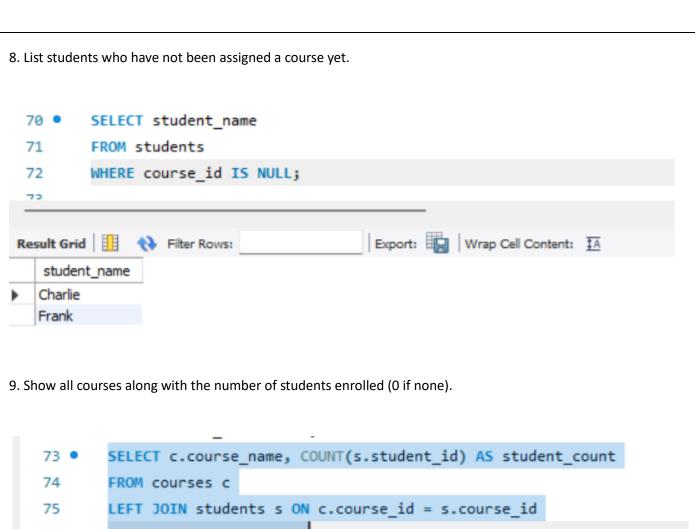
4. List all students whose course credits are greater than 3.

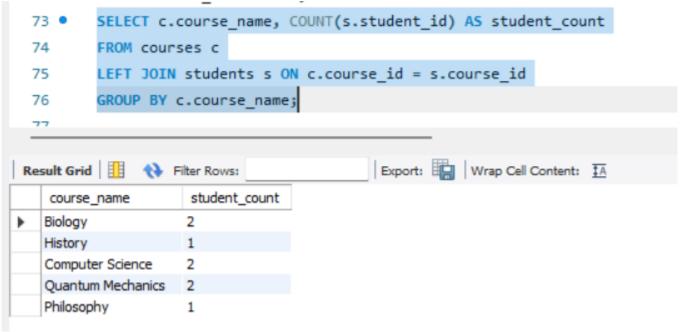


5. Find students who are in the 'Science' department.



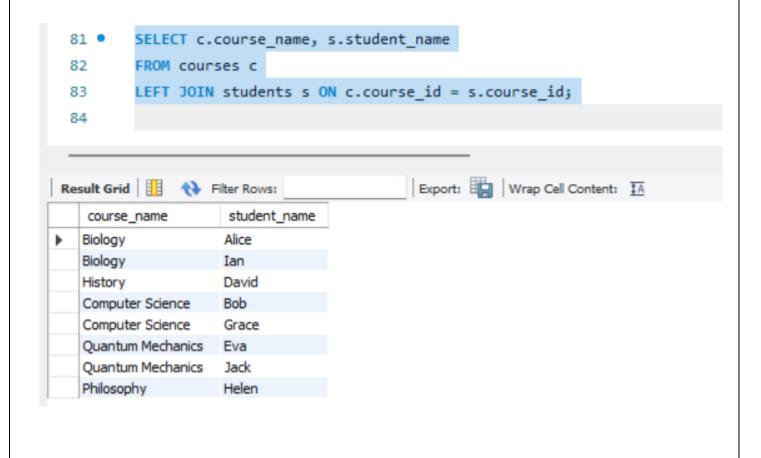




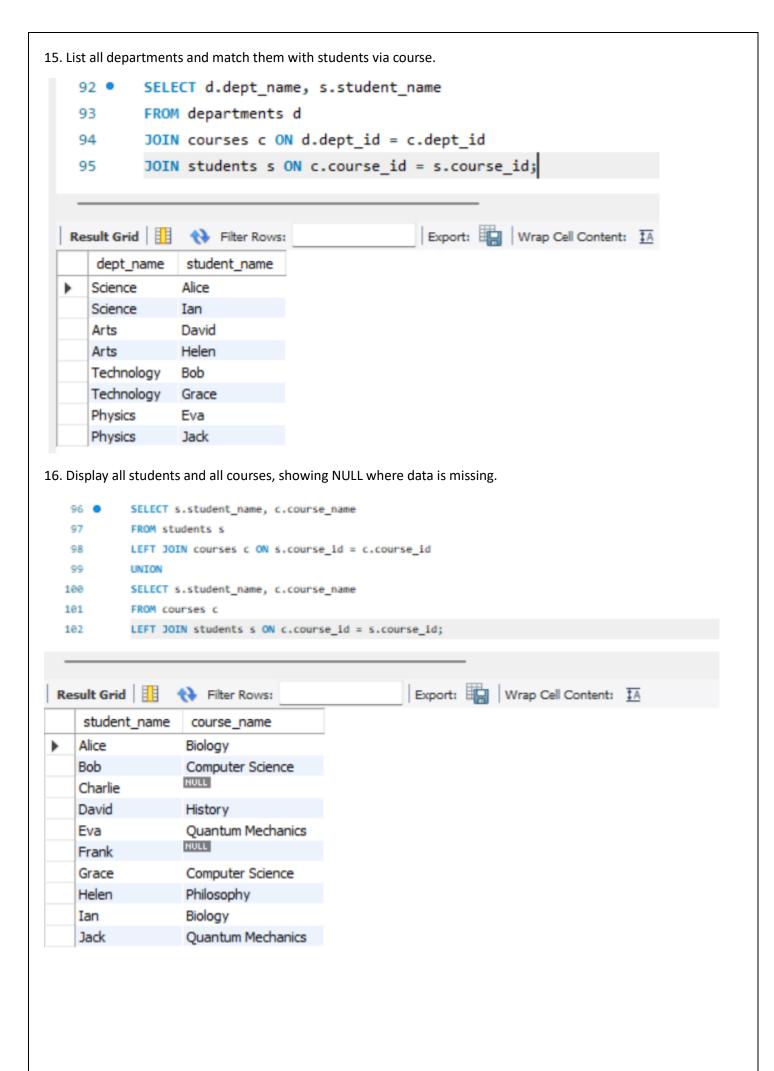


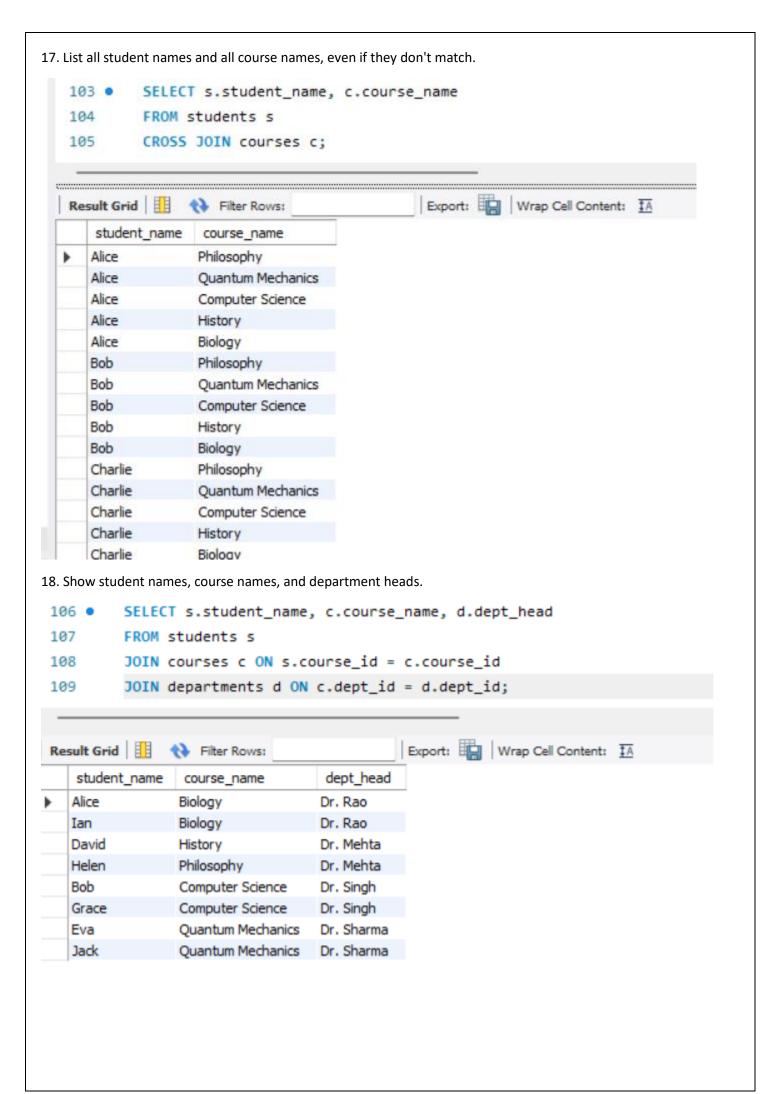
10. Display all students with their course department (NULL if not assigned). SELECT s.student_name, d.dept_name FROM students s 78 LEFT JOIN courses c ON s.course id = c.course id 79 LEFT JOIN departments d ON c.dept id = d.dept id; 80 Export: Wrap Cell Content: IA Result Grid Filter Rows: student_name dept_name Alice Science Bob Technology NULL Charlie David Arts Eva Physics NULL Frank Technology Grace Helen Arts Ian Science Jack **Physics**

11. Display all course names with their student names, even if no students are enrolled.



```
12. Show all departments and the students studying in them, using RIGHT JOIN.
          SELECT d.dept_name, s.student_name
  82
          FROM students s
          RIGHT JOIN courses c ON s.course id = c.course id
  83
          RIGHT JOIN departments d ON c.dept_id = d.dept_id;
  84
                                              Export: Wrap Cell Content: IA
 Result Grid
               Filter Rows:
    dept_name
                student_name
    Science
               Alice
    Science
               Ian
    Arts
               David
    Arts
               Helen
    Technology
               Bob
    Technology
               Grace
    Physics
               Eva
   Physics
               Jack
13. Find courses that do not have any students enrolled.
  85 •
          SELECT c.course name
          FROM courses c
  86
          LEFT JOIN students s ON c.course_id = s.course_id
  87
          WHERE s.student_id IS NULL;
  88
                                              Export: Wrap Cell Content: IA
 Result Grid Filter Rows:
    course_name
14. Show all course names and credits, even if they don't have any students.
            SELECT c.course_name, c.credits
   90
            FROM courses c
            LEFT JOIN students s ON c.course_id = s.course_id;
   91
                                              Export: Wrap Cell Content: IA
  course_name
                       credits
     Biology
                       4
                       4
     Biology
     History
     Computer Science
                       5
     Computer Science
                       5
     Quantum Mechanics 4
     Ouantum Mechanics
                       4
     Philosophy
                       2
```



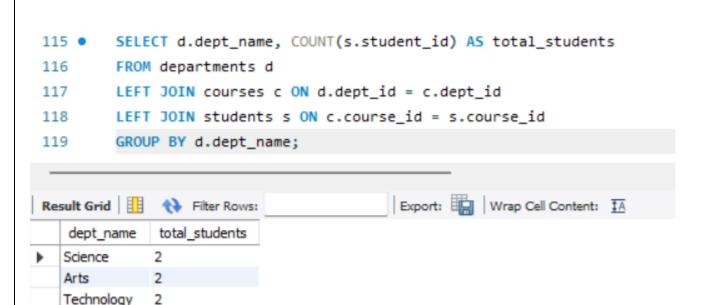


19. Find students who are in courses from the 'Technology' department.

20. Show department names and total students in each department.

Physics

2



```
21. Display student names, their department, and credits for their course.
120 •
          SELECT s.student_name, d.dept_name, c.credits
 121
          FROM students s
 122
          JOIN courses c ON s.course_id = c.course_id
          JOIN departments d ON c.dept_id = d.dept_id;
123
                                             Export: Wrap Cell Content: TA
student name | dept name
                             credits
   Alice
                 Science
                            4
                            4
   Ian
                 Science
   David
                 Arts
                            3
   Helen
                            2
                 Arts
   Bob
                 Technology
                            5
   Grace
                 Technology
                            5
   Eva
                 Physics
                            4
   Jack
                            4
                 Physics
22. Show all departments with courses and students (include those with no students).
124 •
         SELECT d.dept_name, c.course_name, s.student_name
125
         FROM departments d
126
         LEFT JOIN courses c ON d.dept_id = c.dept_id
127
         LEFT JOIN students s ON c.course_id = s.course_id;
                                            Export: Wrap Cell Content: IA
dept_name
              course_name
                                student_name
              Biology
                                Alice
  Science
             Biology
  Science
                               Ian
                               David
  Arts
             History
  Arts
             Philosophy
                               Helen
  Technology
             Computer Science
                               Bob
```

Technology

Physics

Physics

Computer Science

Quantum Mechanics

Quantum Mechanics

Grace

Eva

Jack

23. Find pairs of students studying the same course.

24. Show all possible student pairs where both are in the same department.

```
131 •
        SELECT s1.student_name AS student1, s2.student_name AS student2
132
       FROM students s1
133
        JOIN courses c1 ON s1.course_id = c1.course_id
134
        JOIN departments d1 ON c1.dept_id = d1.dept_id
        JOIN students s2 ON s2.course_id = c1.course_id AND s1.student_id < s2.student_id;</pre>
135
                                      Export: Wrap Cell Content: TA
student1 student2
  Alice
          Tan
  Bob
          Grace
          Jack
  Eva
```

25. List all students who share a course with 'Alice'.

```
FROM students s1

JOIN students s2 ON s1.course_id = s2.course_id

WHERE s1.student_name = 'Alice' AND s2.student_name != 'Alice';

Result Grid Filter Rows:

| Export: | Wrap Cell Content: | IA
```

26. Find the total number of students in each course. SELECT c.course_name, COUNT(s.student_id) AS total_students 141 FROM courses c LEFT JOIN students s ON c.course_id = s.course_id 142 143 GROUP BY c.course_name; Export: Wrap Cell Content: TA course_name total_students Biology History 1 Computer Science 2 2 Quantum Mechanics Philosophy 1 27. Find the average credits for courses that have students enrolled. SELECT AVG(c.credits) AS avg_credits 144 • FROM courses c 145 JOIN students s ON c.course_id = s.course_id; 146 Export: Wrap Cell Content: TA avg_credits 3.8750 28. Show the department with the highest number of enrolled students. 147 SELECT d.dept_name, COUNT(s.student_id) AS total_students FROM departments d 148 JOIN courses c ON d.dept_id * c.dept_id JOIN students s ON c.course_id * s.course_id 150 GROUP BY d.dept name 151 ORDER BY total_students DESC 152 LIMIT 1; 153 Export: Wrap Cell Content: A Fetch rows: dept_name total_students Science 2

