LAB QUESTIONS – MySQL 21st November 2025

PART 1 — SQL Queries

Q1. Create a table named students with fields:

- stdid INT PRIMARY KEY
- stdname VARCHAR(50)
- age INT
- city VARCHAR(50)

Q2. Insert the following records into the students table:

stdid	stdname	age	city
1	Rohan	20	Pune
2	Meera	22	Mumbai
3	Arjun	21	Delhi
4	Kavya	23	Pune
5	Neha	22	Kolkata

Q3. Display all student records.

Q4. Display only the name and age of all students.

Q5. Display students who are from Pune.

Q6. Display students whose age is greater than 21.

Q7. Display students in descending order of age.

Q8. Count how many students belong to each city. (Use GROUP BY)

Q9. Display students whose name starts with 'K'. (Use LIKE)

Q10. Delete student whose stdid = 5.

PART 2 — ALTER COMMAND QUESTIONS

Q11. Add a new column contact VARCHAR(15) to the students table.

Q12. Modify the data type of city column to VARCHAR(100).

Q13. Rename the column stdname to student_name.

Q14. Drop the column contact from the table.

Q15. Add a new column gender ENUM('M','F').

PART 3 — JOIN PRACTICE

Tables:

Table: students

stdid	student_name	city
1	Rohan	Pune
2	Meera	Mumbai
3	Arjun	Delhi
4	Kavya	Pune

Table: marks

stdid	subject	marks
1	Maths	88
2	Maths	76
3	Maths	92
5	Maths	67

INNER JOIN

Q16. Display student name and marks of only those students who have matching IDs in both tables.

(Students without marks should not appear.)

LEFT JOIN

Q17. Display all students and their marks.

(If marks not available, show NULL.)

RIGHT JOIN

Q18. Display all marks records along with student names.

(If student doesn't exist in students table, show NULL.)

CROSS JOIN

Q19. Display all possible combinations of students and subjects.

(Use CROSS JOIN between students and marks table to show every pair.)

JOIN with Filtering

Q20. Using INNER JOIN, display students who scored more than 80.