MySQL Test - 1

Database and Table Management

1. Create a new database called company_db.
2. Select the company_db database for use.
3. Create a table named skills with columns: skill_id (auto-incremented primary key), skill_name (varchar, not null), and category (varchar).
4. Show all databases available on the MySQL server.
5. Show all tables in the company_db database.
6. Rename the employee table to staff.

Data Insertion

7. Insert a new record into the employee table for an employee named "Alice Green" with email "alice.green@company.com", hire date "2024-01-10", salary 62000.00, dept_id 4, and gender "Female".
8. Insert multiple records into the project table: "Mobile App" (budget 60000.00, dept_id 2) and "Training Program" (budget 25000.00, dept_id 1).
9. Add a new department "Sales" located in "Boston" to the department table.
10. Insert a record into the employee table with only first_name ("Tom") and email ("tom@company.com"), leaving other fields as default or NULL.

Data Retrieval

11. Retrieve all records from the employee table.
12. Select only the emp_id, first_name, and salary columns from the employee table.

13. Display emp_id as "Employee ID", first_name as "Name", and email as "Email Address" from the employee table.
14. Retrieve all employees hired after January 1, 2023.
15. List all projects with a budget greater than 40000.00, ordered by budget in descending order.
16. Show the distinct locations from the department table.

Data Modification

17. Add a new column phone_number (varchar, 15 characters) to the employee table after the email column.
18. Update the salary of "John Doe" to 65000.00 in the employee table.
19. Set the gender of all employees in the IT department (dept_id 2) to "Other".
20. Drop the phone_number column from the employee table.

Filtering and Conditions

21. Retrieve employees with a salary between 60000 and 80000.
22. List employees whose first_name starts with "J".
23. Find all projects where the dept_id is either 1 or 2.
24. Show employees whose email is not NULL.
25. Retrieve departments that are not located in "New York" or "Chicago".
26. List employees hired in the year 2023 using the hire_date column.

Aggregate Functions

27. Calculate the total salary of all employees.
28. Find the average budget of all projects.
29. Determine the highest salary in the employee table.

30. Count the number of employees in the IT department (dept_id 2).
31. Find the minimum budget among all projects.

Joins

32. Retrieve all employees along with their department names.
33. List all departments and the number of employees in each, even if a department has no employees.
34. Show all projects along with the department names they belong to.
35. Find employees who work in departments located in "San Francisco".
36. List departments that have no projects assigned to them.

String and Numeric Functions

37. Concatenate first_name and last_name of employees with a space between them as "Full Name".
38. Convert the dept_name column in the department table to uppercase.
39. Extract the first 3 characters of each employee’s email.
40. Calculate the absolute value of -50000.
41. Round the average salary of employees to 2 decimal places.

Advanced Queries

42. Limit the employee list to the first 3 records, ordered by hire_date descending.
43. Retrieve the second page of employees (records 4-6) when paginating with 3 records per page, ordered by emp_id.
44. Use IF to classify employee salaries as "High" if >= 70000, otherwise "Low".

45. Use CASE to categorize project budgets: "Large" (>= 60000), "Medium" (>= 40000), "Small" (< 40000).
46. Find the total budget for projects in each department, grouped by dept_id.

Bonus Questions (Optional)

47. Create a query to find the employee with the longest first_name using LENGTH.
48. Retrieve all employees whose hire date is within the last 90 days from the current date (March 22, 2025).

Deletion and Cleanup

49. Delete all employees with a salary less than 60000.
50. Drop the project table from the database.