Day 1

1. A company stores information about employees, their departments, and project assignments. Convert the following non-normalized table into a normalized form by breaking it into multiple tables to eliminate redundancy and ensure data integrity.

Employee_ID	Employee_Name	Department_ID	Department Name	Project_ID	Project Name
1	Alice	D101	HR	P001	Recruitment
2	Bob	D102	IT	P002	Software Dev
1	Alice	D101	HR	P003	Training
3	Charlie	D103	Sales	P004	Market Analysis

2. A company stores information about products, their suppliers, and orders. Convert the following non-normalized table into a normalized form by breaking it into multiple tables to eliminate redundancy and ensure data integrity.

Product ID	Product_Name	Supplier_ID	Supplier Name	Order_ID	Order_Date
101	Laptop	S001	Tech Supplies	O001	2024-01-15
102	Smartphone	S002	Mobile World	O002	2024-01-16
101	Laptop	S001	Tech Supplies	O003	2024-01-17
103	Tablet	S003	Gadget Zone	O004	2024-01-18

These questions are just a starting point, feel free to explore more.

Day 3

1. Data Definition Language (DDL)

1.1 Create a table named **Employee** with the following fields:

- **Employee_ID** (INT, Primary Key)
- **Employee_Name** (VARCHAR(100))
- **Department** (VARCHAR(50))
- **Salary** (DECIMAL(10,2))
- **Join_Date** (DATE)
- **Age** (INT)

1.2 Alter the **Employee** table to add a new column:

- **Email** (VARCHAR(100))

1.3 Change the name of the column **Employee_Name** to **Full_Name**.

1.4 Remove the **Email** column from the **Employee** table.

1.5 Drop the **Employee** table.

2. Data Manipulation Language (DML)

2.1 Insert the following employee records into the **Employee** table:

- Employee_ID = 101, Employee_Name = 'Alice', Department = 'HR', Salary = 50000, Join_Date = '2020-01-15', Age = 30
- Employee_ID = 102, Employee_Name = 'Bob', Department = 'IT', Salary = 60000, Join_Date = '2021-03-22', Age = 28
- Employee_ID = 103, Employee_Name = 'Charlie', Department = 'IT', Salary = 75000, Join_Date = '2022-05-10', Age = 35
- Employee_ID = 104, Employee_Name = 'Dave', Department = 'HR', Salary = 55000, Join_Date = '2019-07-05', Age = 40

- Employee_ID = 105, Employee_Name = 'Eve', Department = 'Marketing', Salary = 45000, Join_Date = '2021-11-23', Age = 25

2.2 Update the salary of 'Alice' by increasing it by 10%.

2.3 Delete the record of 'Bob' who left the company.

3. Transaction Control

3.1 Set a savepoint called **before_update** before making any updates.

3.2 Commit the changes after inserting and updating the employee records.

3.3 Rollback to the **before_update** savepoint if any update was made incorrectly.

These questions are just a starting point, feel free to explore more.

Day 4

Employee_ID	Employee_Name	Department	Salary	Join_Date	Age
101	Alice	HR	50000	2020-01-15	30
102	Bob	IT	60000	2021-03-22	28
103	Charlie	IT	75000	2022-05-10	35
104	Dave	HR	55000	2019-07-05	40
105	Eve	Marketing	45000	2021-11-23	25
106	Frank	IT	70000	2020-12-01	32
107	Grace	Marketing	46000	2022-03-18	27

1. Write a query to find all employees whose salary is greater than 55000.

2. Write a query to find employees who joined before 2021.

3. Write a query to find all employees whose salary is less than 50000 **and** are **not** older than 30.

4. Write a query to find all employees who are **not** in the "HR" department **and** are older than 30.

5. Write a query to find the average salary of employees in the "IT" department.

6. Write a query to find the total number of employees who are older than 30.

7. Write a query to find the name of the employee with the highest salary in the "Marketing" department.

8. Write a query to find employees whose salary is between 45000 and 70000 and are from either "HR" or "IT" departments.

9. Write a query to find the department with the highest average salary.

10. Write a query to find the second-highest salary from the employee table.

11. Find the employee who joined the company most recently.

12. Write a query to find the salary of all employees after adding a 10% bonus.

For more questions follow : sql queries

These questions are just a starting point, feel free to explore more.

Day 5

Tables:

Employees:

Employee_ID	Employee_Name	Department_ID	Salary
101	Alice	D1	50000
102	Bob	D2	60000
103	Charlie	D1	75000
104	Dave	D3	55000
105	Eve	D2	45000

Departments:

Department_ID	Department_Name
D1	HR
D2	IT
D3	Marketing
D4	Sales

1. Write a query to retrieve the **Employee_Name**, **Department_Name**, and **Salary** of all employees who are assigned to a department (only those with a matching **Department_ID** in both tables).

2. Write a query to retrieve all **Employee_Name** and **Department_Name** where even employees without a department are included. If an employee does not belong to any department, the **Department_Name** should be NULL.

3. Write a query to retrieve the **Employee_Name** and **Department_Name** for all departments, including departments with no employees. If a department has no employees, the **Employee_Name** should be NULL.

These questions are just a starting point, feel free to explore more.