# Lab2

**Title: Data Manipulation Language (DML)**

**Objective:**  
To practice and implement Data Manipulation Language commands.

This lab focuses on core DML operations using SQL, including inserting, updating, selecting, and deleting data. Students will create an **employee** table and perform practical tasks like bulk record insertion, conditional updates, and filtered queries.

**Questions:**

Create a database Lab2 and a table employee with the following structure:

| **Column Name** | **Data Type** |
| --- | --- |
| E\_ID | INT |
| E\_NAME | VARCHAR(20) |
| E\_JOB | VARCHAR(20) |
| E\_SALARY | INT |

**SQL Queries**

**1. Create Database and Table**

CREATE DATABASE Lab2;

USE Lab2;

CREATE TABLE employee (

    E\_ID INT,

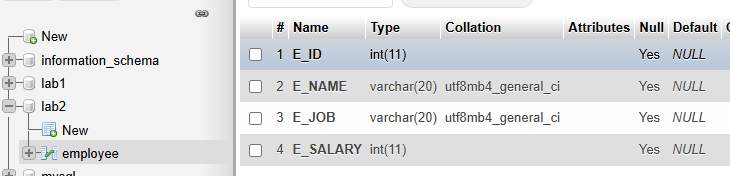
    E\_NAME VARCHAR(20),

    E\_JOB VARCHAR(20),

    E\_SALARY INT

);

**Output:**



**2. Insert a single record in the employee table**

**SQL Queries**

INSERT INTO employee (E\_ID, E\_NAME, E\_JOB, E\_SALARY)

VALUES (1, 'Santosh', 'Programmer', 30000);

**Output:**  


**3. Insert more than one record in the employee table using a single insert command.**

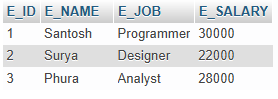
**SQL Queries**

INSERT INTO employee (E\_ID, E\_NAME, E\_JOB, E\_SALARY)

VALUES

    (2, 'Surya', 'Designer', 22000),

    (3, 'Phura', 'Analyst', 28000);

**Output:**  


**4. Update the employee table to set salary of all employees to Rs. 25,000 who work as programmer.**

**SQL Queries**

UPDATE employee

SET E\_SALARY = 25000

WHERE E\_JOB = 'Programmer';

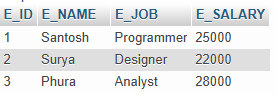
**Output:**  


**5. Select all information from the employee table.**

**SQL Queries**

SELECT \* FROM employee;

**Output:**

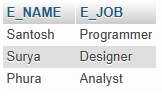


**6. Select employee name (e\_name) and job (e\_job) from the employee table.**

**SQL Queries**

SELECT E\_NAME, E\_JOB FROM employee;

**Output:**

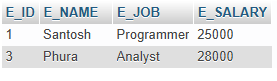


**7. Delete employees working as designer.**

**SQL Queries**

DELETE FROM employee

WHERE E\_JOB = 'Designer';

**Output:**  


**8. Display only employees who work as analyst.**

**SQL Queries**

SELECT \* FROM employee

WHERE E\_JOB = 'Analyst';

**Output:**



**9. Display employee job (e\_job) from the table while avoiding duplicate values.**

**SQL Queries**

SELECT DISTINCT E\_JOB FROM employee;

**Output:**



**10. Display employee name (e\_name) and job (e\_job) for employees with a salary ≥ Rs. 20,000.**

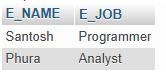
**SQL Queries**

SELECT E\_NAME, E\_JOB

FROM employee

WHERE E\_SALARY >= 20000;

**Output:**

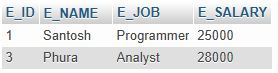


**Appendix: Final Table State**

**SQL Queries**

SELECT \* FROM employee;

**Output:**



**Conclusion**

This lab successfully demonstrated key DML operations for manipulating database records, including inserting, updating, querying, and deleting data. The exercises provide a strong foundation for real-world database administration and data-driven application development.

# sheet

**Objective**

**To practice and implement Data Manipulation Language commands.**

**Lab Exercise**

**Create a Database named** Lab2 **and a table called** employee **with the following structure:**

| **Column Name** | **Data Type** |
| --- | --- |
| e\_id | int |
| e\_name | varchar(20) |
| e\_job | varchar(20) |
| e\_salary | int |

**Tasks**

1. **Insert a single record in the** employee **table.**
2. **Insert more than one record in the** employee **table using a single insert command.**
3. **Update the** employee **table to set salary of all employees to Rs. 25,000 who work as** programmer.
4. **Select all information from the** employee **table.**
5. **Select employee name** (e\_name) **and job** (e\_job) **from the** employee **table.**
6. **Delete employees working as** designer.
7. **Display only employees who work as** analyst.
8. **Display employee job** (e\_job) **from the table while avoiding duplicate values.**
9. **Display employee name** (e\_name) **and job** (e\_job) **for employees with a salary ≥ Rs. 20,000.**

# Lab report answer.

CREATE TABLE EMPLOYEE(

E\_ID INT,

E\_NAME VARCHAR(20),

E\_JOB VARCHAR(20),

E\_SALARY INT

);

INSERT INTO employee (E\_ID, E\_NAME, E\_JOB, E\_SALARY)

VALUE (1, 'SANTOSH', 'DOCTOR', 1000),

(2, 'SARASWOTI', 'VETNARY', 1001);

UPDATE employee SET E\_SALARY=25000 WHERE E\_JOB='DOCTOR' ;

SELECT E\_ID FROM employee WHERE E\_ID='1';

SELECT E\_ID, E\_NAME, E\_JOB, E\_SALARY FROM employee WHERE E\_ID='1';

DELETE FROM employee WHERE E\_JOB='DOCTOR';

// DOCTOR BHAYE KO SABAI.

SELECT DISTINCT