

Name: Prashant Suresh Shirgave.

Roll No: 3

Batch: T1

Class: TY(CSE-AIML)

Experiment No. 4

Title: DML statements

Objective: To study select, from, where clauses & update, delete, insert statements.

Theory:

DML Statements:

DML is an abbreviation of **Data Manipulation Language**.

The DML commands in Structured Query Language change the data present in the SQL database. We can easily access, store, modify, update and delete the existing records from the database using DML commands.

Four main DML commands in SQL:

1. SELECT Command
2. INSERT Command
3. UPDATE Command
4. DELETE Command

1. SELECT Command:

SELECT is the most important data manipulation command in Structured Query Language. The SELECT command shows the records of the specified table. It also shows the particular record of a particular column by using the WHERE clause.

Syntax: SELECT columnlist from tablename [where condition] ;

2. INSERT Command

INSERT is another most important data manipulation command in Structured Query Language, which allows users to insert data in database tables.

Syntax: Insert into tablename [(column1, ... , column n)] values (value1, , value n);

3. UPDATE Command:

UPDATE is another most important data manipulation command in Structured Query Language, which allows users to update or modify the existing data in database tables.

Syntax: update tablename set column1= new value[,column2 = new value,...][where condition];

4. DELETE Command :

DELETE is a DML command which allows SQL users to remove single or multiple existing records from the database tables.

This command of Data Manipulation Language does not delete the stored data permanently from

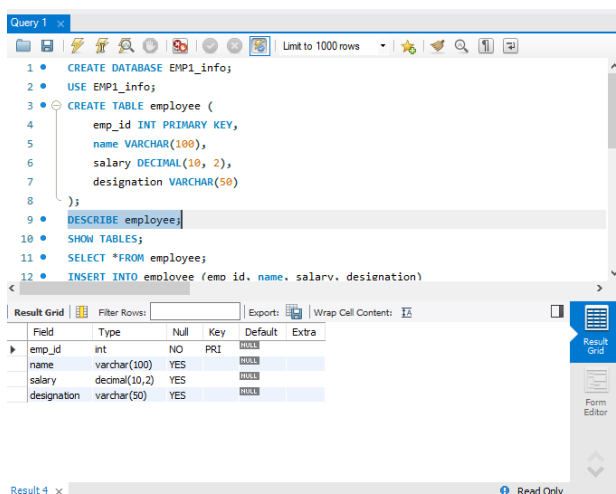
the database. We use the WHERE clause with the DELETE command to select specific rows from the table.

Syntax: delete [from] tablename [where condition];

Consider the following schema

employee(emp_id, name, salary, designation)

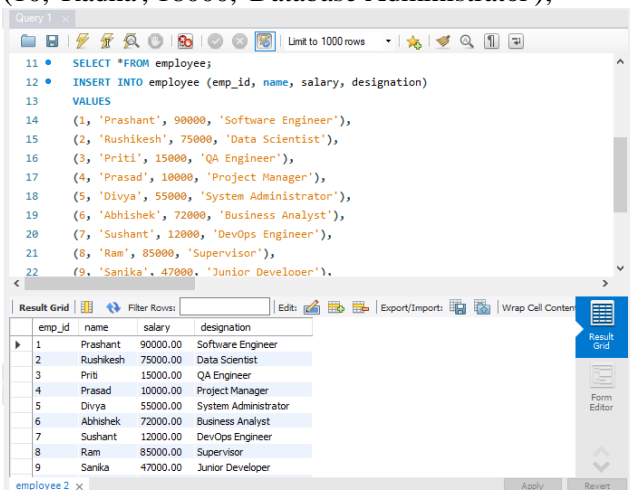
```
CREATE TABLE employee (  
    emp_id INT PRIMARY KEY,  
    name VARCHAR(100),  
    salary DECIMAL(10, 2),  
    designation VARCHAR(50)  
);
```



1. Insert at least 10 records with meaningful data.

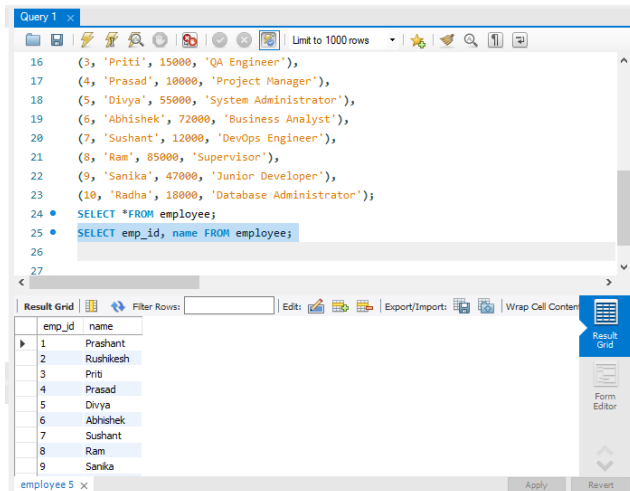
INSERT INTO employee (emp_id, name, salary, designation)
VALUES

- (1, 'Prashant', 90000, 'Software Engineer'),
- (2, 'Rushikesh', 75000, 'Data Scientist'),
- (3, 'Priti', 15000, 'QA Engineer'),
- (4, 'Prasad', 10000, 'Project Manager'),
- (5, 'Divya', 55000, 'System Administrator'),
- (6, 'Abhishek', 72000, 'Business Analyst'),
- (7, 'Sushant', 12000, 'DevOps Engineer'),
- (8, 'Ram', 85000, 'Supervisor'),
- (9, 'Sanika', 47000, 'Junior Developer'),
- (10, 'Radha', 18000, 'Database Administrator');



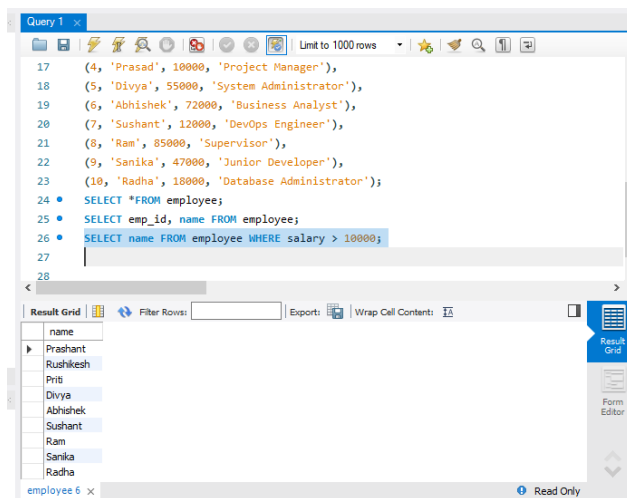
2. Find the name of the employee along with their id.

```
SELECT emp_id, name FROM employee;
```



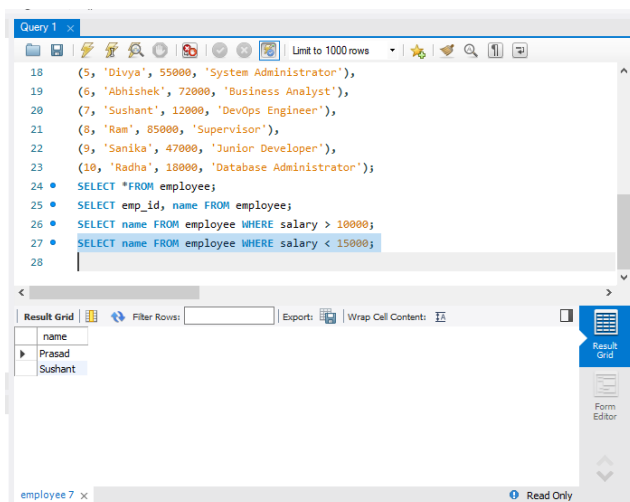
3. Find name of the employees whose salary is >10,000.

```
SELECT name FROM employee WHERE salary > 10000;
```



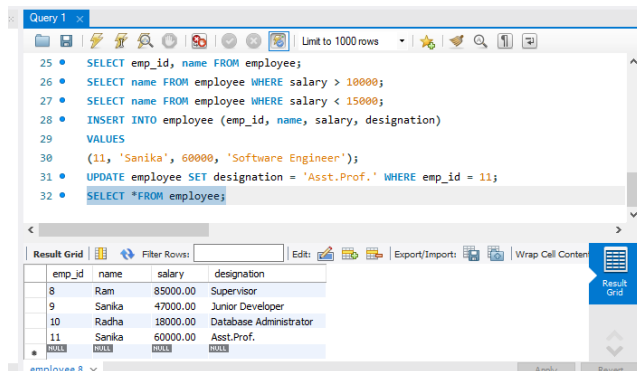
4. Find name of the employees whose salary is <15,000.

```
SELECT name FROM employee WHERE salary < 15000;
```



5. Update designation of employee 11 to 'Asst.Prof.'

UPDATE employee SET designation = 'Asst.Prof.' WHERE emp_id = 11;



The screenshot shows a database query editor with the following SQL commands:

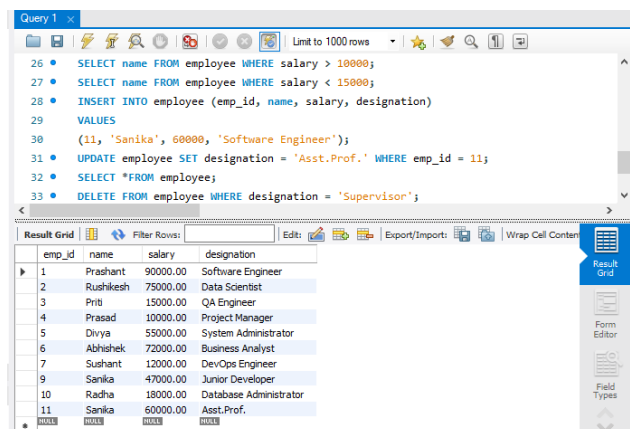
```
25 • SELECT emp_id, name FROM employee;
26 • SELECT name FROM employee WHERE salary > 10000;
27 • SELECT name FROM employee WHERE salary < 15000;
28 • INSERT INTO employee (emp_id, name, salary, designation)
29 VALUES
30 (11, 'Sanika', 60000, 'Software Engineer');
31 • UPDATE employee SET designation = 'Asst.Prof.' WHERE emp_id = 11;
32 • SELECT *FROM employee;
```

The result grid displays the following data:

emp_id	name	salary	designation
8	Ram	85000.00	Supervisor
9	Sanika	47000.00	Junior Developer
10	Radha	18000.00	Database Administrator
11	Sanika	60000.00	Asst.Prof.

6. Delete the employees having designation supervisor.

DELETE FROM employee WHERE designation = 'Supervisor';



The screenshot shows a database query editor with the following SQL commands:

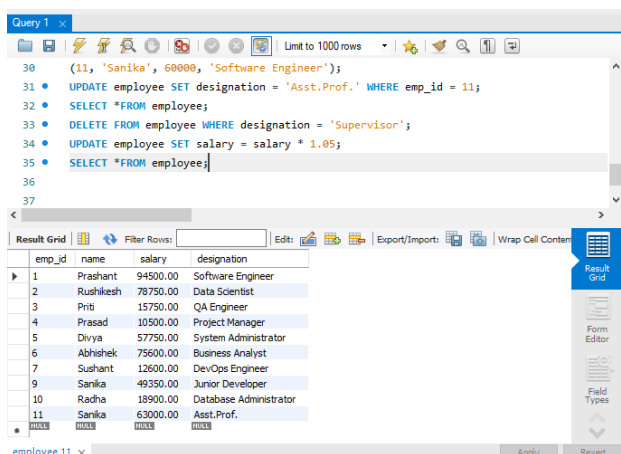
```
26 • SELECT name FROM employee WHERE salary > 10000;
27 • SELECT name FROM employee WHERE salary < 15000;
28 • INSERT INTO employee (emp_id, name, salary, designation)
29 VALUES
30 (11, 'Sanika', 60000, 'Software Engineer');
31 • UPDATE employee SET designation = 'Asst.Prof.' WHERE emp_id = 11;
32 • SELECT *FROM employee;
33 • DELETE FROM employee WHERE designation = 'Supervisor';
```

The result grid displays the following data:

emp_id	name	salary	designation
1	Prashant	90000.00	Software Engineer
2	Rushikesh	75000.00	Data Scientist
3	Priit	15000.00	QA Engineer
4	Prasad	10000.00	Project Manager
5	Divya	55000.00	System Administrator
6	Abhishek	72000.00	Business Analyst
7	Sushant	12000.00	DevOps Engineer
9	Sanika	47000.00	Junior Developer
10	Radha	18000.00	Database Administrator
11	Sanika	60000.00	Asst.Prof.

7. Increment the salary of employees by 5%.

UPDATE employee SET salary = salary * 1.05;



The screenshot shows a database query editor with the following SQL commands:

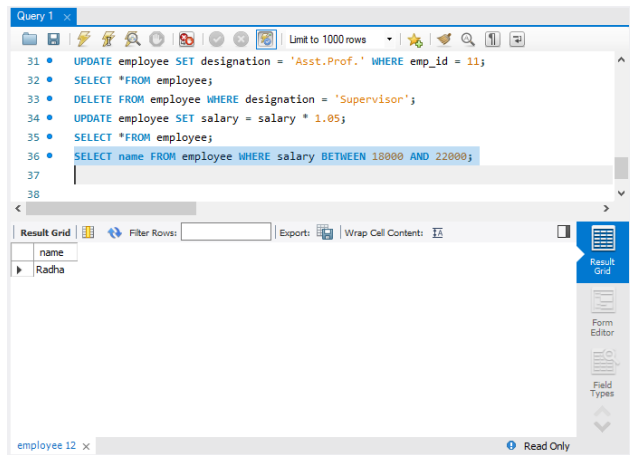
```
30 • (11, 'Sanika', 60000, 'Software Engineer');
31 • UPDATE employee SET designation = 'Asst.Prof.' WHERE emp_id = 11;
32 • SELECT *FROM employee;
33 • DELETE FROM employee WHERE designation = 'Supervisor';
34 • UPDATE employee SET salary = salary * 1.05;
35 • SELECT *FROM employee;
```

The result grid displays the following data:

emp_id	name	salary	designation
1	Prashant	94500.00	Software Engineer
2	Rushikesh	78750.00	Data Scientist
3	Priit	15750.00	QA Engineer
4	Prasad	10500.00	Project Manager
5	Divya	57750.00	System Administrator
6	Abhishek	75600.00	Business Analyst
7	Sushant	12600.00	DevOps Engineer
9	Sanika	49350.00	Junior Developer
10	Radha	18900.00	Database Administrator
11	Sanika	63000.00	Asst.Prof.

8.Find the name of employees having salary between 18000 and 22000.

SELECT name FROM employee WHERE salary BETWEEN 18000 AND 22000;



Outcome: Students are able to manipulate the database using DML statements.

