Prajakta Patil Name:-

DBMS

Lab Assignment - 3

- Title:- Design at least 10 SQL queries for suitable database application using SQL DML statement. Insert, select, Update, Delete with operations. functions & set operator

- Roll No :- CO3048

- Class :- TE

- Subject :- Database Management System Laboratory

Title : F
Selec

2) Insert
Purpose - Adds new rows to a table.
Syntax:
INSERT INTO table name (column1, column2, ...)
VALUES (1, John Doe', Software Engineer, '60000);
eg
To add a new employee to the employee tables
INSERT INTO table name
VALUES (value 1, value 2)

3) Update.
Purpose: Modifies existing rows in a table
syntax:
UPDATE table_name
SET column 1 = value 1, column 2 = value 2, ...
WHERE Condition;
eg
To updata the salary of the employee tables
INSERT INTO employee (id name, position, salary)
VALUES (1, John Doe, software Engineer 6000);

4) DELETE:
Purpose :- Removes rows from table
syntax:
DELETE FROM table_name
WHERE Condition;

Aggregation functions:

Aggregat^{n} fu^{n} in SQL used to perform calculations on multiple rows of data to return a single summary value.

1) Count :

Purpose: Return the no of rows that match a specified condition.

Syntax - COUNT (expression)

eg.

```
SELECT COUNT (*) As total_employee
FROM employees.
```

2) SUM:

Purpose: Return the total sum of a numeric column

syntax: SUM column_name

eg.

```
SELECT SUM (salary) As total_salary
FROM employee.
```

3) AVG:

Purpose: Return the avg value of numeric column

syntax: AVG (column_names)

eg.

```
SELECT AVG (salary As average_salary)
FROM employees.
```

- Aim :- Design atleast 10 SQL queries for suitable database application using SQL DML statm Insert, select, Update, Delete with opercn, fun & set operator.

- Objective :-
  To develop basic intermediate & advanced database programming skills.

- Theory :-

DML Statements
- Data Manipulatn Language statements are a subject of SQL commands used to manage & manipulate the data stored within a relational database.
- DML statements allowed to perform various operatn on the data, such as retrieving inserting updating & deleting records.

1) SELECT
- Purpose : Retrieves data from one or more tables
- syntax:

```
SELECT column1, column2, ...
FROM table_name WHERE condition
```

- eg.
  Suppose we have a table name employees with column id, name, position & salary.

* Comparison operator :-

1) Equal to (=) :-
Purpose :- checks if two values are equal
syntax :- column_name = value
eg.
SELECT * FROM employee, WHERE name = Alice';

2) NOT equal to ('<>' or '!=')
Purpose :- checks if two values are not equal
syntax :- column name <> value
eg.
SELECT * FROM employee
WHERE department <> Sales

3) Greater than (>) :-
Purpose :- checks if the value on the left is greater than the value on right
syntax :-
column_name > value

4) Less than (<)
Purpose :- checks if the value on the left is less than the value on right
syntax :-
column_name < value

Title :
Selec

4) MIN
Purpose - Return the smallest value in a column
syntax: MIN (column name)
eg
SELECT MIN (salary) As Lowest_salary
FROM employees.

5) MAX:
Purpose - Return the largest value in a column
syntax: MAX (column-name)
eg
SELECT MIN (salary) as lowest_salary
FROM employees

5) Max
Purpose - Return the largest value in a column
syntax - MAX (column-name)
eg
SELECT MAX (salary) As highest_salary
FROM employees;

Eg

To sort employee By salary in descending order

```
SELECT employee ID, Name, salary
FROM employees
ORDER By salary DESC;
```

△

☆ Conclusion:-

Thus we have studied how to use & implement DML statement & write queries

5) Greater than equal to (>=)
Purpose - checks if the values on left is greater than or equal to value on right
syntax :-

column_name >= value

6) less than or equal to (<=)
Purpose checks if the value on left is less than or equal to value on right.
syntax.

Column_name <= value

- Order by clause :

- It is clause in SQL used to sort the result of query in either ascending or descending order of a query based on one or more columns
By default it sorts in ascending order
syntax :

SELECT column 1, column 2, ---
FROM table_name
ORDER By Column 1 [AS DESC]
Column 2 [AS DESC] ---