Date:
26/08/2025

Task-3 - Using clauses apertures & Functions

Aim: To implement the DML commands using clauses, operators & functions in queries.

1] Insert Into

```
Insert into team values (1, warrior, "stadium A",
                         Null);
```

```
>> select * from teams;
```

TeamID	Name	Homeground	Coach_ID
1	warriors	Stadium A	NULL

2) UPDate

```
Update Team set 'warriors' = 'Mafia' where
teamID = 1;
```

TeamID	Name	Homeground	Coach ID
1	Mafia	Stadium A	NULL

3) Delete

Delete from team where TeamID = 1;

4) Truncate: It removes data but structure remains.

Select Query:

Select CoachID, TeamID where homeground = 'stadium A';

Coach ID	TeamID
10210	1

→ Select Name from Team where TeamID = 1;

Name
warriors

→ Select Homeground from Team where TeamID=1;

Home ground
Stadium A
Stadium B
stadium E

VEL TECH	
EX NO.	3.1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	15
SIGN WITH DATE	19/8/25

Result: The implementation of DML commands in sql for sports management system has been done successfully.

Date: 26/8/25 Task 3.2 Aggregate Functions (Multi-row ops)

Aim: To ~~study~~ implement ~~and~~ implement aggregate funcs (count(), sum(), Avg(), min(), max()) on a sports management system.

Commands

1) count(): It counts the no. of rows.

>>> select count(*) as Name from Team;

Output:

Total Name
11

2) Max(): Finds highest amount in row.

» select max(Name) from players;

Total Name
242

3) Avg(): Finds average from row.

>>> Select Avg (Experience) from ~~Chennai~~.Coach;

Average Experience
112

4) Min(): Finds Minimum value from row.

» select min(Experience) from Coach;

Output:

min Experience
10

5) Sum(): Adds the values from row.

» select min(Team Name) from Team;

Sum Team Name
22

VEL TECH	
EX No.	3.2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
TOTAL (20)	15
SIGN WITH DATE	26/8/23

Result: The implementations of Aggerate func. has been done successfully