19/8/25 TASK: 1 Using Clauses, operators and Functions in Queries

Aim: To implement the DML commands using clauses, operators and function in SQL

To understand the different issues involved in the design and implementation of a database System.

DML commands:

1. INSERT 2. UPDATE. 3. DELETE

1. INSERT INTO:

Inserting a single record.

```
INSERT INTO student values(103, 'Ritheesh', 'ritheesh@gmail.
com', 18, 2023, 103, 125)
```

std-id	name	email	age	academic-year	dept-id	slot-id
101	Arun	arun@gmail.com	20	2022	101	123
105	Ram	Ram@gmail.com	19	2021	105	125
103	Ritheesh	ritheesh@gmail.com	18	2023	103	125

```
INSERT INTO Slot values(127, 'AN', 'vikram', 2020-05-19,
'F30');
```

Slot-id	slot-type	Name	join-date	venue
127	AN	vikram	2020-05-19	F30.
128	FN	vinay	2020-06-20	F30

2. UPDATE SET WHERE:

update value in particular record.

Slot-id	Slot-type	Name	Join_Date	Venue
127	AN	vikram	2020-05-19	F30
128	AN	vinay	2020-06-20	F30

3. Delete_From:

Delete all the records with the table.

Delete_from_where:

Delete specific record from the table.

DELETE_FROM slot where slot_id = '128'

Slot-id	Slot-type	Name	Join_Date	Venue
127	AN	vikram	2020-05-19	F30

4. Truncate:

Delete ~~the~~ all ~~tables~~ records from the table but not the structure.

Truncate Table ~~department~~ Slot;

5) Like(%):

Retrieve the name end with any charecter letter using %, if (%n) last charecter.

select name From Student where name like "~~R~~m%"

Name
Ram
Ritheesh

6) between , range:

Select *From Student where' Student_id between 100 and 102;

Std-id	Name	email	Age	Academic year	Slot-id	dept-id
101	Arun	arun@gmail.com	20	2022	123	101

7) select * From student where student_id >= 101

Std-id	Name	email	Age	Academic-year	slot-id	dept-id
101	Arun	arun@gmail.com	20	2022	123	1

8) Select distinct academic-year from student.

Academic_year
2022
2021
2023

9) ~~Union~~ Minimum:

select * min ~~lo~~(Age) from Student;

Age
18

VEL TECH	
EX NO.	3.1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	-
TOTAL (20)	18
SIGN WITH DATE	

19/8/2

Result.

of

26/8/25 Task 3 : Aggregate Functions (Multi-Row Operation)

Aim:
To ~~Study and~~ implement aggregate function count(), sum(), Avg(), min(), Max() on a sample Student database.

Procedure: 1. Create table named credits
2. Insert sample records.
3. Write queries using aggregate function.
4. Observe and record output.

Table credits:

Std-id	credits
101	42
102	44
103	46

commands:

1) count total numbers & rows:
select count(*) from credits;

count(*)
3

2) Higest credits obtained by student;
select MAX(credits) from Credits;

MAX(credits)
46

Avg (credits)
44

Avg (credits) calculates mean value of all credits.

4) ~~Fin~~ Minimum credits among students

select min (credits) from credits;

min (credits)
42

min (credits) find the lowest credits.

5) Find total credits obtained by students.

select sum (credits) from credits;

sum (credits)
132

sum (credits) add up all credits in column credits

VEL TECH	
EX NO.	32
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
TOTAL (20)	15
SIGN WITH DATE	26/8/25

Result:

Of

Implements the ~~oo~~ DML comm