TASK No:3 DML Commands using clauses, operators and functions in queries

23/8/25

Aim :- To implement DML Commands using clauses, operators and functions in queries.

DATA MANIPULATION LANGUAGE (DML);

the data manipulation language (DML) is used to retrieve, insert and modify database information. these commands will be used by all database users during the routine operation of the database, let's take a brief look at the basic DML commands,

1, INSERT 2, UPDATE 3, DELETE

1, INSERT INTO; this is used to add records into a relation. these are three type of insert int que~~stion~~ries which are as

Inserting a single record

Syntax; INSERT INTO <relation/table name> (field-1, field 2 ... fiel

VALUES (data-1, data-2, data-n)'

Example; SQL>insert into member values (116,'shan','CSE','male');

inserting a single record

→Member

membno	name	department	gender
111	Jeni	IT	female
111	Jaz	CSE	female
113	John	CSE	female
114	BEn	CSE	female
115	Ann	CSE	female
116	shan	CSE	male

→Borrowed

1. output :-

Name
John
Ben
Ann

2. output :-

isbn	title	authors	Pagecount	Publisher
1111	operating system	Silberschatz	750	MCgraw

2. UPDATE - SET - WHERE; this is used to update the content of a record in a relation,

Syntax; SQL>UPDATE relation name SET field - name 1= data, field - name 2 = data = 'Kumar' WHERE Sno=1'

3. DELETE - FROM; this is used to delete all the records of a relation but it will retain the structure of that relation,

a) DELETE - FROM; this is used to delete all the records of relation

Syntax; SQL>DELETE FROM relation-name;

Example; SQL>Delete From Std;

b) Delete - FROM - WHERE; this is used to delete a selected record from a relation,

Syntax; SQL>DELETE FROM relation-name WHERE condition

Example; SQL>DELETE FROM Student WHERE Sno=2;

5. TRUNCATE; this command will remove the data permanently. But structure will not be removed,

Difference between Truncate & Delete

By using truncate command data will be removed permanetly & will not get back where as by using delete command and data,

SAMPLE queries and output:-

→ 1. Retrieve member name end with letter 'n' and member no between 111 and 115,

Query:

SELECT first-name, last-name, salary FROM employees WHERE first-name LIKE '%m';

→ 2. list books where page count between 700 and 800-

between 700 and 800 - between clause, and operator.

Query;

select* from book where pagecount between 700 and 800;

3. output

min(pagecount)
330

4. output

membno	isbn	issuedate
113	1110	2023 02-07

5. output

membno
11
113
114
115
116

⑥ output

membno
117
113
114
115
116
117
121

⑦ output

Membno
111
111
116
117

⑧ output :-

authors	publisher	NO
MAriA allen	springer	1
sliberschatz	Mcgraw	1
sliberchatz	springer	1
william stalings	ferrouzon	1

→3.
find the authors who has minimum number of page count

[illegible] rate

Query:

select min(page count) from books;

→4. find the records issue date greater than or equal to 02.02.23

Query:

select * from borrowed where issue date >= '2023-02-07';

→5.
list the members, but the same members are listed ones.

Query:

select members distinct member from member;

→6. Combine the records of member and books relation-union

Query:

Select memberno FROM member UNION SELECT memberno FROM

→7. Groupby the member number based on their gender and department.

Query:

SELECT memberno FROM Member GROUP By gender, department;

→8.
find the authors and their publication details using groupby and orderby clauses,

Query:

SELECT authors publisher, COUNT (*) no FROM book GROUP BY authors, publisher order by authors.

VEL TECH	
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
	15

28/8/25

Result:-
the task to implement the DML commands are executed successfully.

→3.
find the records who has minimum number of Pagecount

Aggregate

Query:

Select min(Page count) from books;

→4. find the records issue date greater than or equal to 07-02-23

Query:

select * from borrowed where issuedate >='2023-02-07';

→5.
list the membno, but the same membno are listed ones.

Query:

select ~~mem~~ distinct membno from member;

→6. Combine the records of member and books relation-union

Query:

Select membno FROM member UNION SELECT membno FROM

→7. Groupby the member number based on their gender and department.

Query:

SELECT membno FROM Member GROUP By gender, department;

→8.
find the authors and their publication details using groupby and orderby clauses,

Query:

SELECT authors publisher, COUNT (*) ~~[illegible]~~ FROM book GROUP BY authors, publisher order by authors;

VELTECH	
PERFORMANCE (5)	5
RECORD AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
TOTAL	15

28/8/25

RESult:-
the task to implement the DML commands are executed success

TASK:- 3.2 AGGREGATE FUNCTIONS (MULTI-ROW OPERATIONS)

9/03/25

Aim :-

To study and implement aggregate function (COUNT (), SUM(), AVG(), MIN( ), MAX( )) on a sample student database.

PROCEDURE

1. create a table named students.
2. insert sample records.
3. write queries using aggregate functions.
4. observe and record the output.

COMMANDS with EXPLANATION

1) Count the total number of students

```
SELECT COUNT (*) AS Total-students
FROM students;
```

Explanation:

- COUNT (*) counts how many rows (students) are in the table
- AS Total-students gives a user-friendly column name.

2) find the highest marks obtained by a student

```
SELECT MAX(marks) AS Highest-Mark
FROM student;
```

Explanation:

- Max (Marks) returns the maximum value in the marks column.
- ~~AS~~ this tells us the top scorer's mark.

3. find the average marks of students

```
SELECT AVG (Marks) AS Average-Mark
FROM Students;
```

Explanation:

- AVG (Marks) calculates the mean (average) of all student marks.

Select * From Student 2

Roll No	NAME	AGE	DEPT ID	Mark
1	Arjun	20	101	85
2	Snela	21	101	90
3	Ravi	19	102	75
4	Chandu	18	102	95
5	Shiva	16	101	60
6	naveen.	15	100	80

4) find the minimum mark among students in the ECE department

```
SELECT MIN(MARKS) AS Min_ECE Mark
FROM Students
WHERE Department = 'ECE';
```

Explanantion;

- MIN(MARKS) finds the lowest mark,
- WHERE Department = 'ECE' restricts the calculation only to ECE students.

5. find the total marks scored by students in each department

```
SELECT Department, SUM (Marks) AS Avg - Marks
FROM Students
GROUP BY Department;
```

Explanation;

- SUM (marks) adds up marks,
- GROUP BY Department ensures that the total is calculated for each department separately.

6. find the average marks per department, ordered by average marks descending

```
SELECT Department, AVG(MARKS) AS AVG-MARKS
FROM Students
GROUP BY Department
ORER BY AVG-MARKS DESC;
```

Explanation;

- AVG(MARKS) gives the average per department
- GROUP BY Department groups students by their department
- ORDER BY AVG_MARKS DESC sorts results so that departments with the highest average come first

VEL TECH	
EX.NO.	
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	15
SIGN WITH DATE	8/08/25

2/8

Result:- thus the SQL commands Executed Successfully in database management system,