

Task-3t OML commands using clauses operators

Date:- 18/8/25 and function in Queries.

Aim:- To implement of OML commands using clauses, operators and functions in querying.

Data manipulation Language (OML):-

The Data manipulation language (OML) 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 look at the basic OML commands.

1. INSERT
2. UPDATE
3. DELETE

INSERT INTO

This is used to add records into a relation

Syntax:-

INSERT INTO <table-name> Field₁, Field₂,
Field N)

values (data₁, data₂ --- data_N),

Example:-

SOL

INSERT INTO patients values ('111', 'Arun',
'Cardiology', 'Male'))

Table after INSERT:-

Patient ID	Patient Name	Department	Gender
111	Arun	cardiology	Male

UPDATE — SET WHERE:-

This is used to update the content of a record in a relation.

Syntax:-

SQL

UPDATE table_name SET field_1 = data WHERE condition;

Example:-

SQL

UPDATE Patients SET PatientName = 'kumar'
WHERE Patients ID = 111;

Table after UPDATE

Patient ID	Patient Name	Department	Gender
111	kumar	cardiology	Male

DELETE From:-

This is used to delete all records of a relation but if retains the structure.

Syntax:-

SQL

DELETE From table-name;

Example

SQL

DELETE From Appointments;

Appointments false after Delete;

Appointment ID Patient ID Doctor ID Appoint. surface
ment

DELETE - FROM - WHERE

this is used to delete greater records from a relation.

Syntax:-

~~SQL~~
DELETE From table-name WHERE condition
Example:

Savli

DELETE FROM Doctors WHERE DoctorID = 202;

Doctors table after DELETE

Doc for ID	Doc for Name	Department	Fees
201	Dr. Sharma	cardiology	1000
203	Dr. Ahmed	Neurology	900
204	Dr. Rajesh	orthopedic	500
205	Dr. Neha	Dermatology	300

TRUNCATE:-

This removes all data permanently but keeps the table structure.

Syntax:-

SQL

TRUNCATE TABLE table-name;

Example:-

SQL

TRUNCATE TABLE Patients;

Patients table after TRUNCATE:

Patient ID Patient Name Department Gender.

Sample queries and outputs:-

"Retrieve patient names ending with letter 'n' and patient no b/w 111 and 115"

Query:-

SQL

SELECT Patient Name, Department, gender
FROM Patients

WHERE Patient Name LIKE '%n' AND Patient ID
BETWEEN 111 AND 115;

Patient Name	Department	Gender
Arun	cardiology	male
Karan	orthopedics	male
Robin	Dermatology	male

2. List doctors whose consultation fees are below 700 and 800

Query:

SQL

SELECT * FROM Doctors WHERE fees Between 700 and 800

Doctor ID	Doctor Name	Department	Fees
202	Dr. Priya	Pediatrics	700
205	Dr. Neha	Dermatology	800

3. Find the record with minimum appointment duration

Query

SQL

SELECT MIN(Duration) FROM Appointment;
MIN(Duration)

20

4. Find appointments with date \geq '2025-2-7'

Query

SQL

SELECT * FROM Appointments WHERE
Appointment Date \geq '2023-02-07'

Appointment ID

302

303

304

305

Patient ID

112

113

114

115

Doctor ID

203

204

202

205

Appointment
month/date

2023-2-7

23-2-9

23-2-10

23-2-12

Duration

45

20

60

25

5. list distinct patient IDs

Query:

SQL:

SELECT DISTINCT Patient ID FROM Patients

Patient ID
111
112
113
114
115

6. combine patient IDs from Patients and
HPP departments (UNION)

Query:

SQL

SELECT Patient ID FROM Patients

UNION

SELECT Patient ID FROM HPP Department

Output:

Patient ID
111
112
113
114
115

7. Group Patients based on gender and
department query:

SQL

SELECT Department, gender, COUNT(*) AS
total patients
FROM Patients

GROUP BY Department, gender;

Department	Gender	Total Patients
Cardiology	Male	1
Neurology	Female	1
Orthopedics	Male	1
Pediatrics	Female	1
Dermatology	Male	1

B. Find doctors and their department details using GROUP BY and ORDER BY

Query:

SQL:

```
SELECT Doctor Name, Department, COUNT(*) AS
Count
FROM Doctors
GROUP BY Doctor Name, Department;
ORDER BY Doctor Name;
```

Doctor Name	Department	Count
Dr. Ahmed	Neurology	1
Dr. Neha	Dermatology	1
Dr. Priya	Pediatrics	1
Dr. Rajesh	Orthopedics	1
Dr. Sharma	Cardiology	1

VELTECH	
DATA MANAGEMENT (5)	3.1
RESULT AND ANALYSIS (5)	5
ALGORITHM (5)	5
CODE (5)	5
TOtal	15

Result for the implementation of OML using triggers, stored procedures, functions executed during session.

Task :- 8.2

Date :- 25/8/25

Aggregate Functions (multi Row operations).

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

Procedure:-

1. Create a table named students.
2. Insert simple records
3. Write queries using aggregate functions
4. Observe and record the output.

COMMANDS WITH EXPLANATION

Patient ID	Patient Name	Department	Bill Amount
101	Arun	cardiology	2000
102	Sneha	Neurology	3500
103	Karan	orthopedics	1500
104	Neena	Pediatrics	4000
105	Tohan	Paediatrics	

1. count the total number of patient

SQL:

SELECT COUNT(*) AS totalPatients
From Patients;
Output;
total - Patients

3. Find the highest bill amount

SQL:

```
SELECT MAX(Bill Amount) AS highest-Bill  
FROM Patients;
```

Output:-

Highest - Bill

4000

3. Find the average bill amount of patients

SQL:

```
SELECT Avg(Bill amount) AS Average - Bill  
FROM Patients;
```

Output:-

Average - Bill

27.00

4. Find the minimum bill amount among patients in neurology department.

SQL

```
SELECT MIN(Bill Amount) AS min-Neuro-Bill
```

Output:-

min - Neuro - Bill

3500

5. Find the total bill amount by each department.

SQl:

```
SELECT Department , sum(Bill Amount) AS  
total - Bill  
FROM patients
```

Group by Department;

Output

Department	Total Bill
cardiology	2000
Neurology	3500
Orthopedics	1500
Pediatrics	4000
Dermatology	

6. Find the average bill for department, ordered by average descending.

SQl:

```
SELECT Department , AVG(Bill Amount) AS Avg  
Bill.  
FROM patients  
GROUP BY Department  
ORDER BY Avg-Bill DESC;
```

outpatient

Department	Aug - 2011
Pediatrics	4000
Neurology	3500
Dermatology	2500
cardiology	2000
orthopedics	1500

VELTECH	
EX No.	3.2
PERFORMANCE (S)	3
RESULT AND ANALYSIS	5
VIVA VOCE (S)	5
RECORD(S)	15
TOTAL (S)	15
SIGN WITH DATE	25/8/25

~~result~~, the implementation of aggregate functions are executed successfully.