

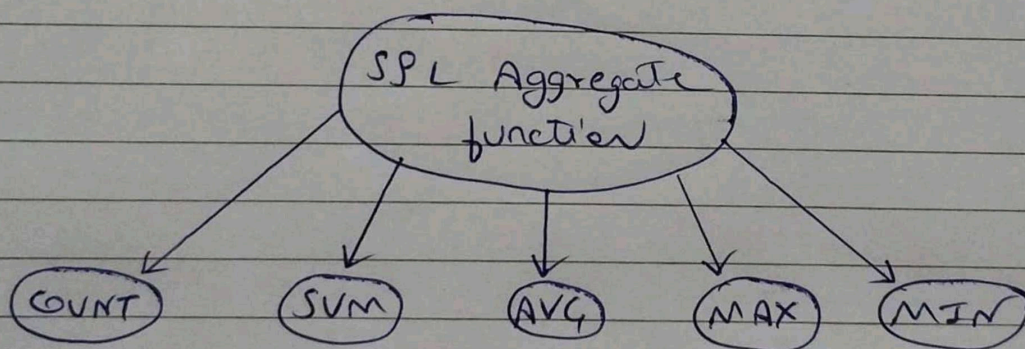
EXPERIMENT - 5

AIM: Implementation of various aggregate functions in SQL.

THEORY :

SQL aggregation function is used to perform the calculation on multiple rows of a single column of a table. It returns a single value. It is also used to summarize the data.

Types of SQL Aggregate function



1) COUNT Function

- COUNT function is used to count the number of rows in a database table. It can work both numeric and non-numeric data types.
- COUNT function uses the COUNT(*) which returns the count of all the rows in a specified table. COUNT(*)

considers duplicate and NULL.

Syntax: COUNT ([ALL | ^{IC}DISTINCT] expression)
OR

COUNT (*)

Example: SELECT COUNT (ename) FROM Employee

2) SUM function

→ SUM function is used to calculate the sum of all selected columns. It works on numeric fields only.

Syntax: SUM() or
SUM (ALL [DISTINCT] expression)

3) AVG Function

→ The AVG function is used to calculate the average value of numeric type. AVG function returns the average of all non-Null values.

Syntax: AVG() OR
AVG [ALL | DISTINCT] expression)

Example: SELECT AVG (Salary)
FROM Emp-Company;

4) MAX Function

→ MAX Function is used to find the maximum value of certain column. This function determines the largest value of all selected values of a column.

Syntax: MAX() (or)
MAX [ALL | DISTINCT] expression)

Example: SELECT MAX (Salary)
FROM Emp-Company;

5) MIN Function

MIN Function is used to find the minimum value of a certain column. This function determines the smallest value of all selected values of a column.

Syntax: MIN() (or)
MIN [ALL | DISTINCT] expression)

Example: SELECT MIN (Salary)
FROM Emp-Company;

INPUT:

```
DROP TABLE IF EXISTS Employee;
DROP TABLE IF EXISTS Emp_Company;
DROP TABLE IF EXISTS Company;
DROP TABLE IF EXISTS Manager;
DROP TABLE IF EXISTS Emp_Shift;

CREATE TABLE Employee(ename varchar2(10),city varchar2(10));
CREATE TABLE Emp_Company(ename varchar2(10),cname varchar2(10),salary
number(7,2),jdate date);
CREATE TABLE Company(cname varchar2(10),city varchar2(10));
CREATE TABLE Manager(ename varchar2(10),mname varchar2(10));
CREATE TABLE Emp_Shift(ename varchar2(10),shift varchar2(10));

INSERT INTO Employee values
('Sunil','Madras'),('Vijay','Madras'),('Amar','Delhi'),('Atharva','Delhi'),('A
mish','Bombay');
INSERT INTO Emp_Company values ('Sunil','ACC',5000,'01-SEP-
2027'),('Vijay','TATA',40000,'01-SEP-2027'),('Amar','Microsoft',6000,'01-SEP-
2027'),('Atharva','Google',60000,'01-SEP-2027'),('Amish','TATA',30000,'02-
SEP-2027');
INSERT INTO Company
values('ACC','Madras'),('TATA','Bengaluru'),('Microsoft','Bombay'),('Google','
Delhi'),('Microsoft','Bombay');
INSERT INTO Manager values
('Sunil','Sharvari'),('Vijay','Sunil'),('Amar','Mahi'),('Atharva','Rucha'),('A
mish','Sam');
INSERT INTO Emp_Shift
values('Sunil','A'),('Vijay','A'),('Amar','B'),('Atharva','C'),('Amish','D');

SELECT * FROM Employee;
SELECT * FROM Emp_Company;
SELECT * FROM Company;
SELECT * FROM Manager;
SELECT * FROM Emp_Shift;
```


DATABASES :

Employee

ename	city
Sunil	Madras
Vijay	Madras
Amar	Delhi
Atharva	Delhi
Amish	Bombay

Company

cname	city
ACC	Madras
TATA	Bengaluru
Microsoft	Bombay
Google	Delhi
Microsoft	Bombay

Emp_Company

ename	cname	salary	jdate
Sunil	ACC	5000	01-SEP-2027
Vijay	TATA	40000	01-SEP-2027
Amar	Microsoft	6000	01-SEP-2027
Atharva	Google	60000	01-SEP-2027
Amish	TATA	30000	02-SEP-2027

Manager

ename	mname
Sunil	Sharvari
Vijay	Sunil
Amar	Mahi
Atharva	Rucha
Amish	Sam

Emp_Shift

ename	shift
Sunil	A
Vijay	A
Amar	B
Atharva	C
Amish	D

Query 1 : List the name of employee having maximum salary.

Code :

```
SELECT ename FROM Emp_company WHERE salary IN (SELECT  
MAX(salary) FROM Emp_company);
```

Output:

ename
Atharva

Query 2 :List the names of employees having maximum salary in their company.

Code :

```
SELECT e1.ename  
FROM Emp_company e1  
WHERE e1.salary IN (SELECT MAX(e2.salary) FROM Emp_company e2 GROUP  
BY e2.cname);
```

Output :

ename
Sunil
Vijay
Amar
Atharva

Query 3 : Find the average salary of each company except 'ACC'.

Code :

```
SELECT cname,AVG(salary) FROM Emp_company WHERE cname!='ACC'  
GROUP BY cname;
```

Output :

cname	AVG(salary)
Google	60000.0
Microsoft	6000.0
TATA	35000.0

Query 4 : Find the average salary of company only for those employees living in 'Delhi'.

Code :

```
SELECT cname,AVG(salary)  
FROM Emp_company  
WHERE ename IN (SELECT ename FROM employee WHERE city='Delhi')  
GROUP BY cname;
```

Output :

cname	AVG(salary)
Google	60000.0
Microsoft	6000.0

Query 5 : Find the name of company having the highest average salary.

Code :

```
SELECT e1.cname
FROM Emp_company e1
WHERE e1.cname=(SELECT cname FROM Emp_company WHERE salary=(SELECT
MAX(salary) FROM Emp_company));
```

Output :

cname
Google

Query 6 : List the number of employees living in 'Bombay'.

Code :

```
SELECT COUNT(ename) FROM employee WHERE city='Bombay';
```

Output :

COUNT(ename)
1

Query 7 : List the name of employees with his living city having maximum salary in compant 'TATA'.

Code :

```
SELECT e1.ename,e1.city
FROM employee e1,Emp_company e2
WHERE e2.salary IN (SELECT MAX(salary) FROM Emp_company WHERE cname='TATA')
and e2.ename=e1.ename;
```

Output :

ename	city
Vijay	Madras

CONCLUSION: Various aggregate functions in SQL are being studied and implemented successfully.