Experiment No. 5

Aim:	To understand & apply aggregate functions such as COUNT, SUM, MIN, MAX & AVG in SQL to analyze student data.
Objective:	Applying aggregate functions like COUNT, AVG, MIN, MAX, SUM in SQL.

Aim:	To understand & apply aggregate functions such as COUNT, SUM, MIN, MAX & AVG in SQL to analyze student data.
Objective:	Applying aggregate functions like COUNT, AVG, MIN, MAX, SUM in SQL.
About the aggregate functions:	i) COUNT : Counts the total number of rows in a Table. eg: SELECT COUNT (*) FROM Student ; ii) SUM : Calculates the sum of values in a column. eg: SELECT SUM (Fees_Paid) FROM Student ; iii) MIN : Finds a minimum value in a column. eg: SELECT MIN (Fees_Paid) FROM Student ; iv) MAX : Finds a maximum value in the column. eg: SELECT MAX (Fees_Paid) FROM Student ; v) AVG : Calculates the average value in a column. eg: SELECT AVG (Fees_Paid) FROM Student ;

Procedure :

I> Create the "Student" Table :

```
CREATE TABLE Student (
    ID INT PRIMARY KEY,
    Name VARCHAR (100),
    Age INT,
    Branch VARCHAR (50),
    FeesPaid DECIMAL (10, 2)
);
```

II> Insert the sample data :

Inserting 5 records :

```
INSERT INTO Student (ID, Name, Age, Branch, Fees Paid)
    VALUES
        (1, 'Bobby', 18, 'CSBS', 50000.00)
        (2, 'Charlie', 18, 'CSBS', 45000.00)
        (3, 'Alice', 19, 'CSBS', 45000.00)
        (4, 'Jane', 18, 'CSBS' 50000.00)
        (5, 'Mona', 17, 'CSBS', 45000.00);
```

III> Apply Aggregate functions :

i> Find total number of students :

```
SELECT COUNT (*) AS Total_Students FROM
                    Students;
```

ii) Find the number of students admitted in CSBS:

```
SELECT COUNT (*) AS CSBS_Students FROM Students
        WHERE BRANCH = 'CSBS';
```

iii) Find the total fees paid by the students:

```
SELECT SUM(Fees_Paid) AS Total_Fees_Paid FROM Students;
```

iv) Find the minimum & maximum fees paid by the students:

```
SELECT MIN (Fees_Paid) AS Min_Fees, MAX (Fees_Paid) AS
        MAX_FEES FROM Student;
```

v) Find the average fees paid by Student:

```
SELECT AVG (Fees_Paid) AS AVERAGE_Fees FROM
        Students;
```

IV) The ~~total~~ table's elements:

Primary_key = ID

NOT NULL = ID, Name, Age, Branch, Fees_Paid

UNIQUE = ID

CHECK = Age

DECIMAL (10, 2) = Fees_Paid

Conclusion : This experiment exercise helps in understanding how aggregate functions (COUNT, SUM, MIN, MAX, AVG) can be used to analyze & summarize database records.

queries.sql + Rushi_Daulatkar AI NEW MYSQL RUN

```sql
-- Step 1: Create the patients Table
CREATE TABLE patients (
    id INT PRIMARY KEY,
    name VARCHAR(50),
    age INT,
    disease VARCHAR(50),
    medical_bill DECIMAL(10,2)
);

-- Step 2: Insert Sample Data
INSERT INTO patients (id, name, age, disease, medical_bill) VALUES
(1, 'Alice', 30, 'Flu', 500.00),
(2, 'Bob', 45, 'Diabetes', 1200.50),
(3, 'Charlie', 28, 'Flu', 450.75),
(4, 'David', 50, 'Heart Disease', 3500.00),
(5, 'Emma', 35, 'Asthma', 800.00);

-- Step 3: Execute Aggregate Functions

-- 1. COUNT() - Total number of patients
SELECT COUNT(*) AS total_patients FROM patients;

-- 2. SUM() - Total revenue from medical bills
SELECT SUM(medical_bill) AS total_revenue FROM patients;

-- 3. AVG() - Average medical bill
SELECT AVG(medical_bill) AS avg_bill FROM patients;

-- 4. MIN() - Minimum medical bill
SELECT MIN(medical_bill) AS min_bill FROM patients;

-- 5. MAX() - Maximum medical bill
SELECT MAX(medical_bill) AS max_bill FROM patients;

-- Step 4: Execute All Aggregate Functions Together
SELECT
    COUNT(*) AS total_patients,
    SUM(medical_bill) AS total_revenue,
    AVG(medical_bill) AS avg_bill,
    MIN(medical_bill) AS min_bill,
    MAX(medical_bill) AS max_bill
FROM patients;
```

Output:

```
+----------------+
| total_patients |
+----------------+
|              5 |
+----------------+
+---------------+
| total_revenue |
+---------------+
|       6451.25 |
+---------------+
+-------------+
| avg_bill    |
+-------------+
| 1290.250000 |
+-------------+
+----------+
| min_bill |
+----------+
|   450.75 |
+----------+
+----------+
| max_bill |
+----------+
|  3500.00 |
+----------+
+----------------+---------------+-------------+----------+----------+
| total_patients | total_revenue | avg_bill    | min_bill | max_bill |
+----------------+---------------+-------------+----------+----------+
|              5 |       6451.25 | 1290.250000 |   450.75 |  3500.00 |
+----------------+---------------+-------------+----------+----------+
```

Conclusion: This experiment exercise helps in understanding how aggregate functions (COUNT, SUM, MIN, MAX, AVG) can be used to analyze & summarize database records.

(A)+ Excellent