College of Engineering, Trivandrum

Department of Computer Science and Engineering

CS333 APPLICATION SOFTWARE DEVELOPMENT LAB

LABORATORY REPORT 5

Aggregate Functions

Student Name	Student ID
1. Justine Biju(S5)	170445(Roll No:37)

Submission Date : 29/07/2019

1 Introduction

SQL has many aggregate functions that help us in performing many mathematical functions. Some of them are listed below:

1. AVG()
   (a) The AVG() function returns the average value of a numeric column.
2. MAX()
   (a) The MAX() function returns the maximum value of a numeric column.
3. MIN()
   (a) The MIN() function returns the minimum value of a numeric column.
4. COUNT()
   (a) The COUNT() function returns the number of rows that matches a specified criteria.
5. SUM()
   (a) The SUM() function returns the total sum of a numeric column.

All these SQL queries are supported by PostgreSQL and can be implemented in a similar fashion.

2 Implementation in PostgreSQL

Let's take a look at the database we are going to work with.

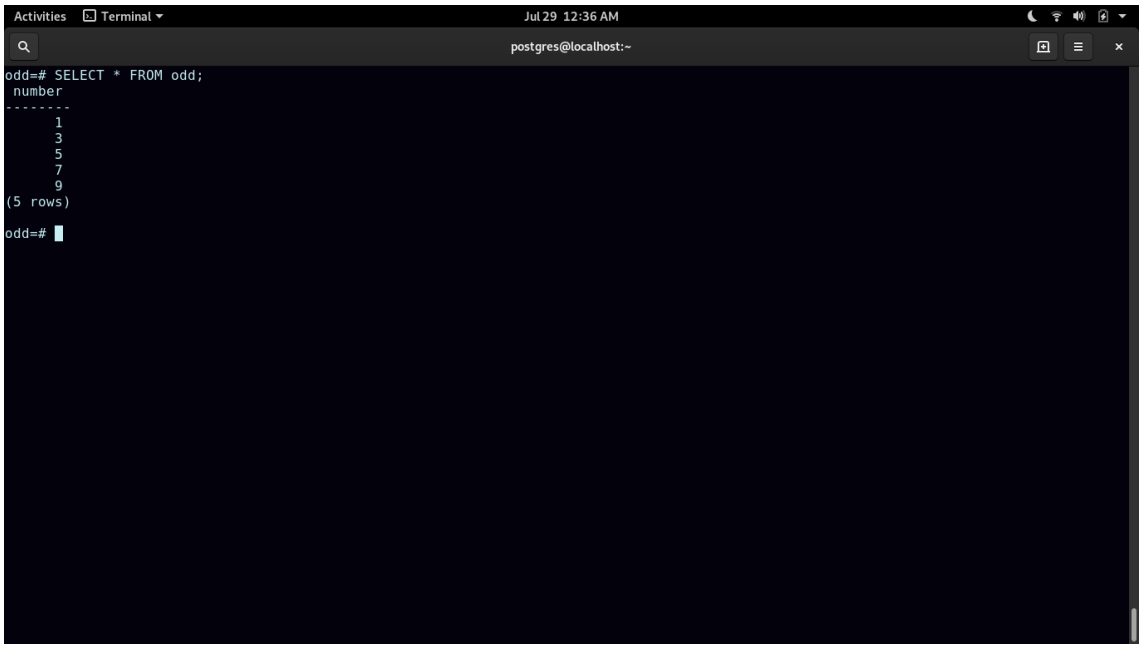

Figure 1: Entities in the table 'odd'

1. Lets find the average of the given numbers using the AVG() function. This can be done using the following query:

```
SELECT AVG(number)
FROM odd;
```

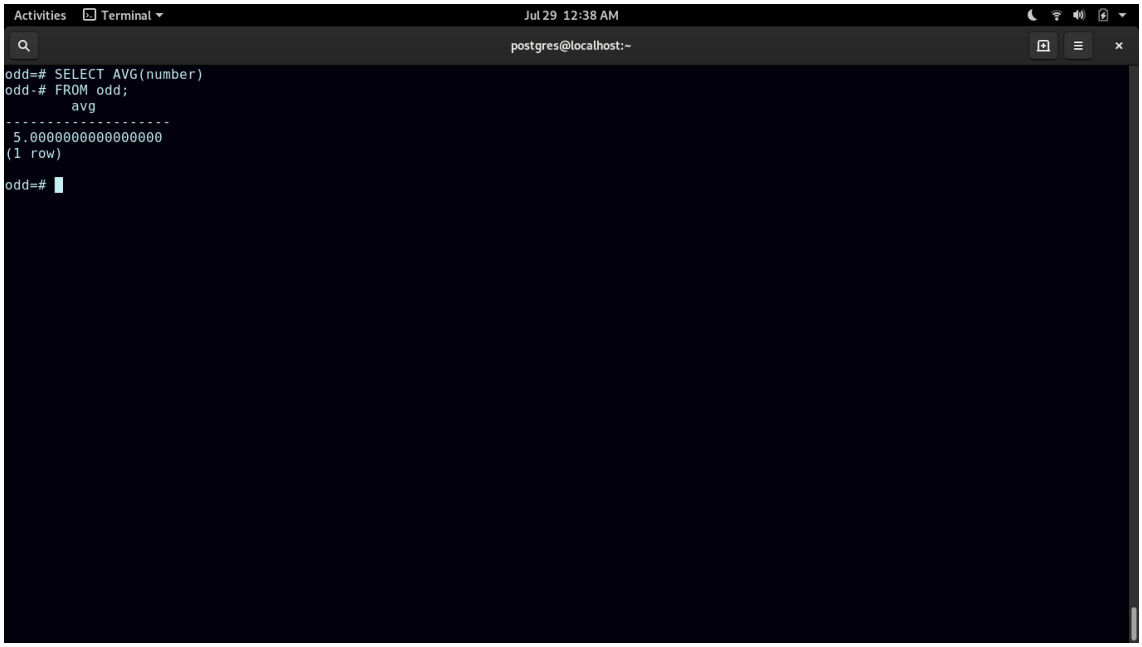

Figure 2: Using AVG() function

2. Now lets find the largest number using MAX() function.

```
SELECT MAX(number)
FROM odd;
```

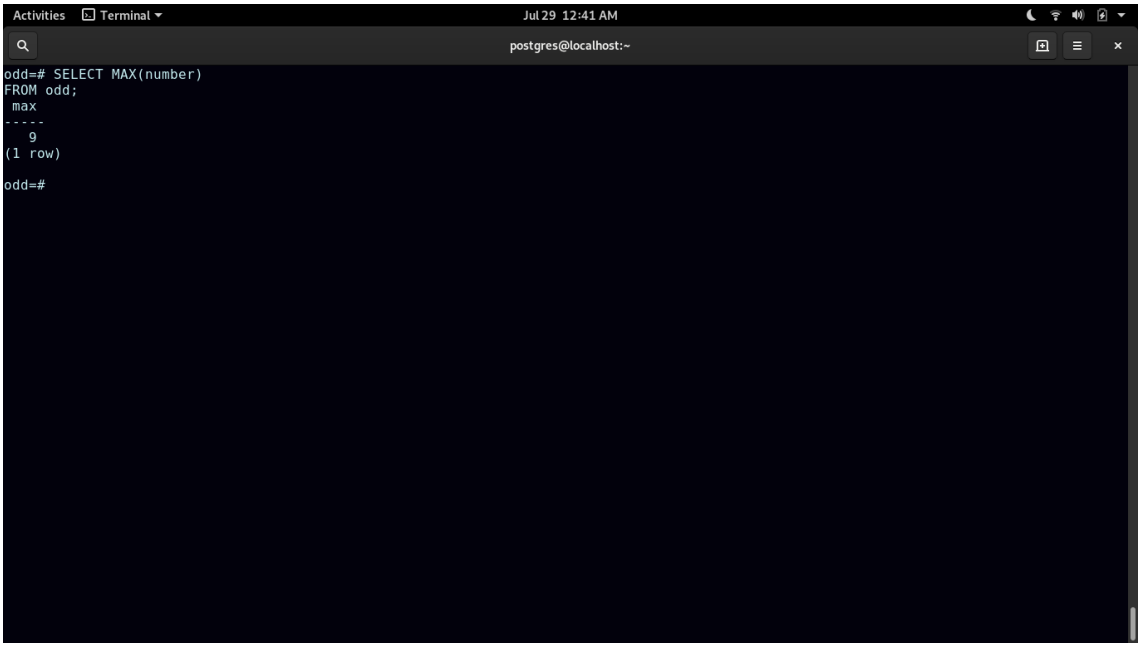

Figure 3: Using MAX() function

3. Similarly lets use the MIN() function to find the smallest number.

```
SELECT MIN(number)
FROM odd;
```

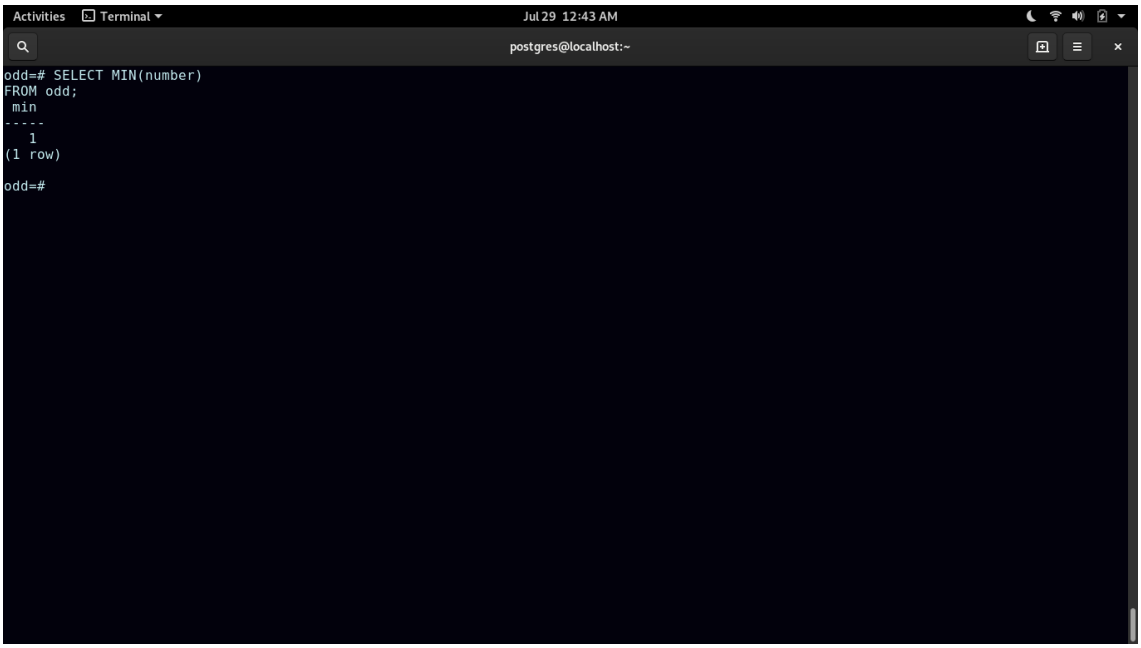

Figure 4: Using the MIN() function

4. COUNT() function can be used to count the number of NON NULL values in a column

```
SELECT COUNT(number)
FROM odd;
```

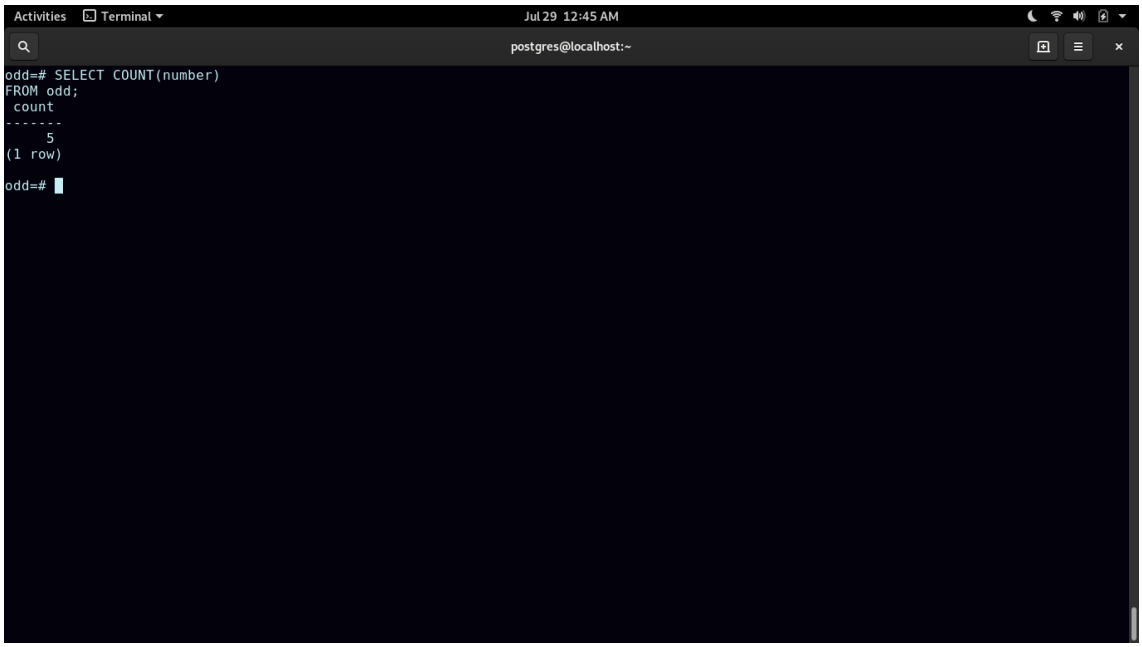

Figure 5: Using the COUNT() function

5. Lets find the sum of the numbers present in the database using the SUM() function.

```
SELECT SUM(number)
FROM odd;
```

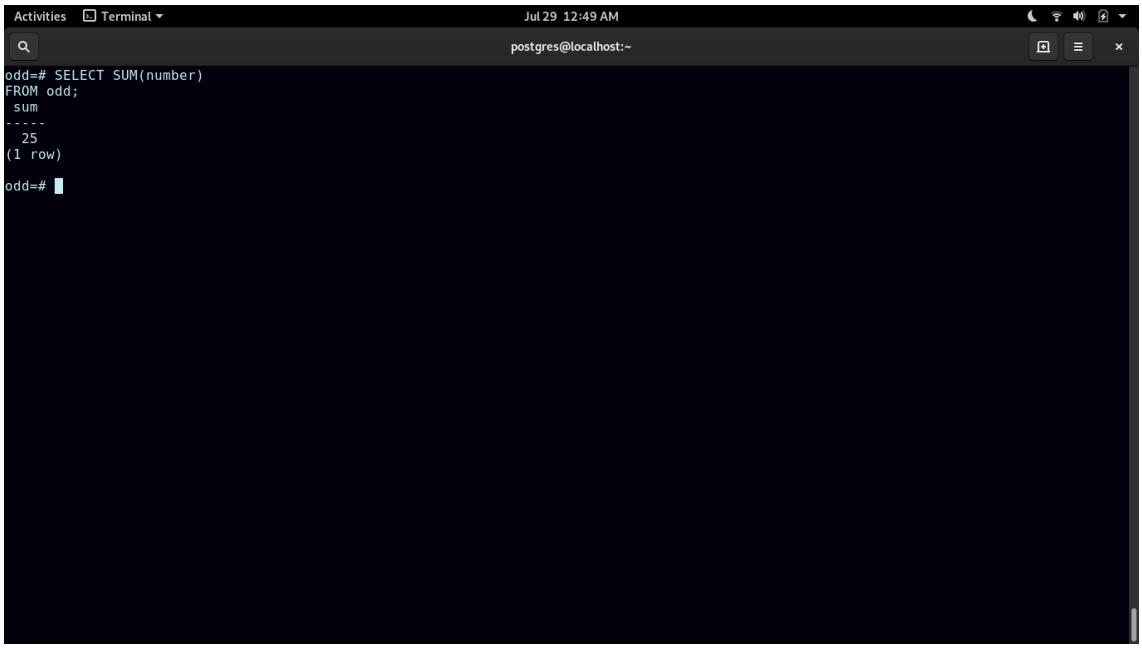

Figure 6: Using SUM() function

3 Questions

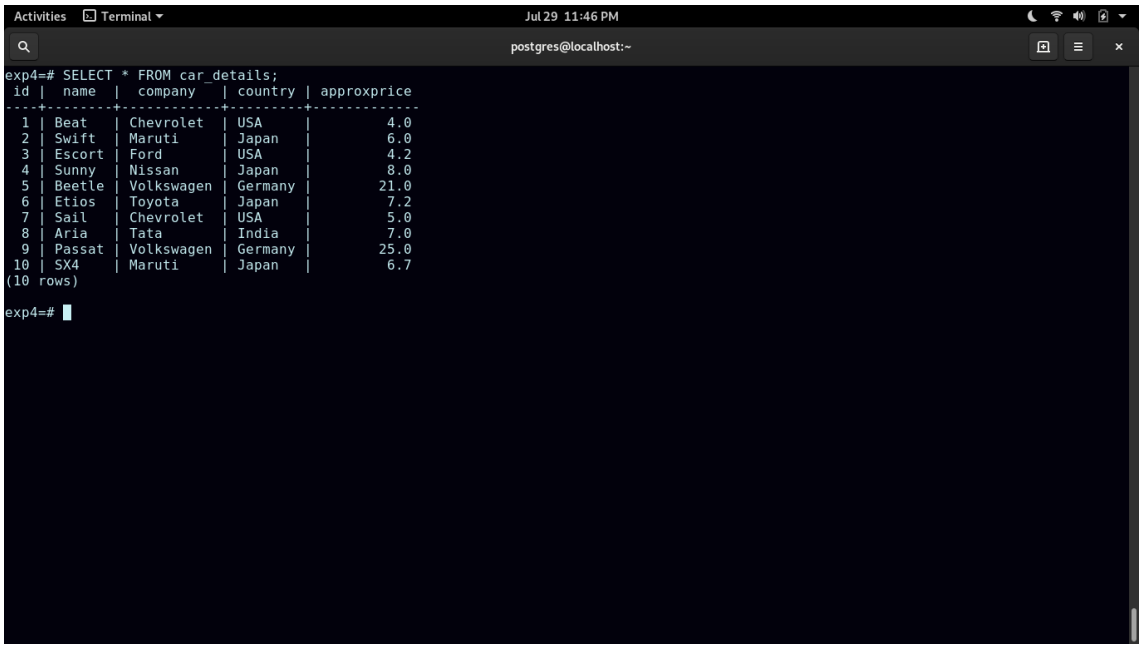

Figure 7: Entries in 'student'

1. Find the class average for the subject 'Physics'

```
SELECT AVG(physics)
FROM student;
```

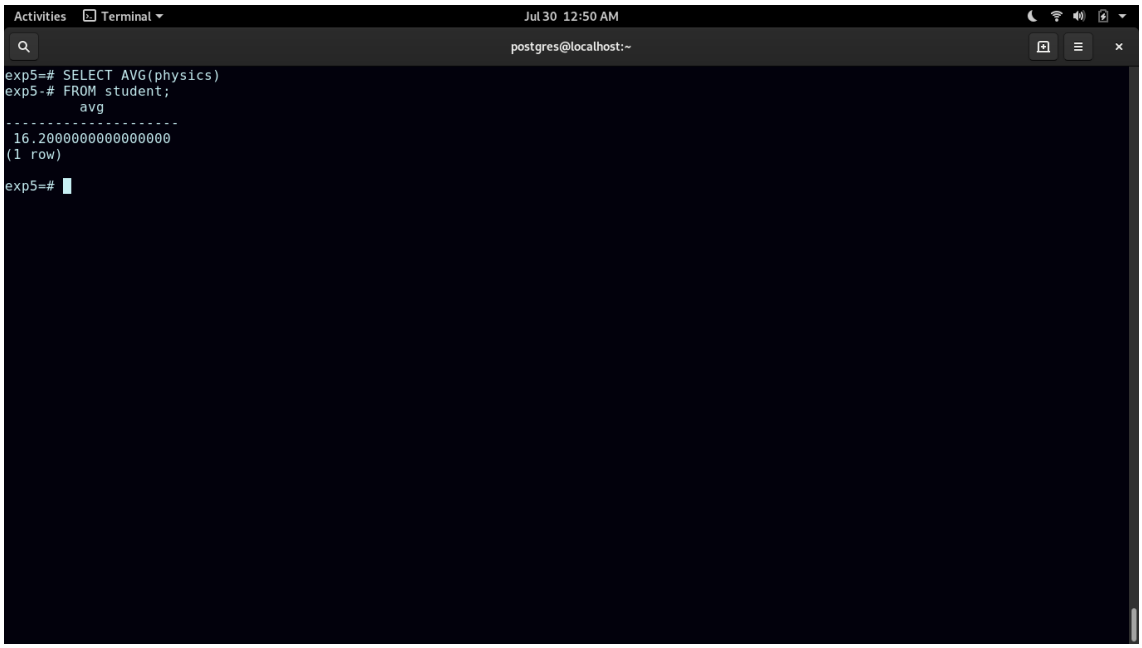

Figure 8: Question 1

2. Find the highest marks for mathematics (To be displayed as highest_marks_maths).

```
SELECT MAX(maths) AS
highest_marks_maths
FROM student;
```

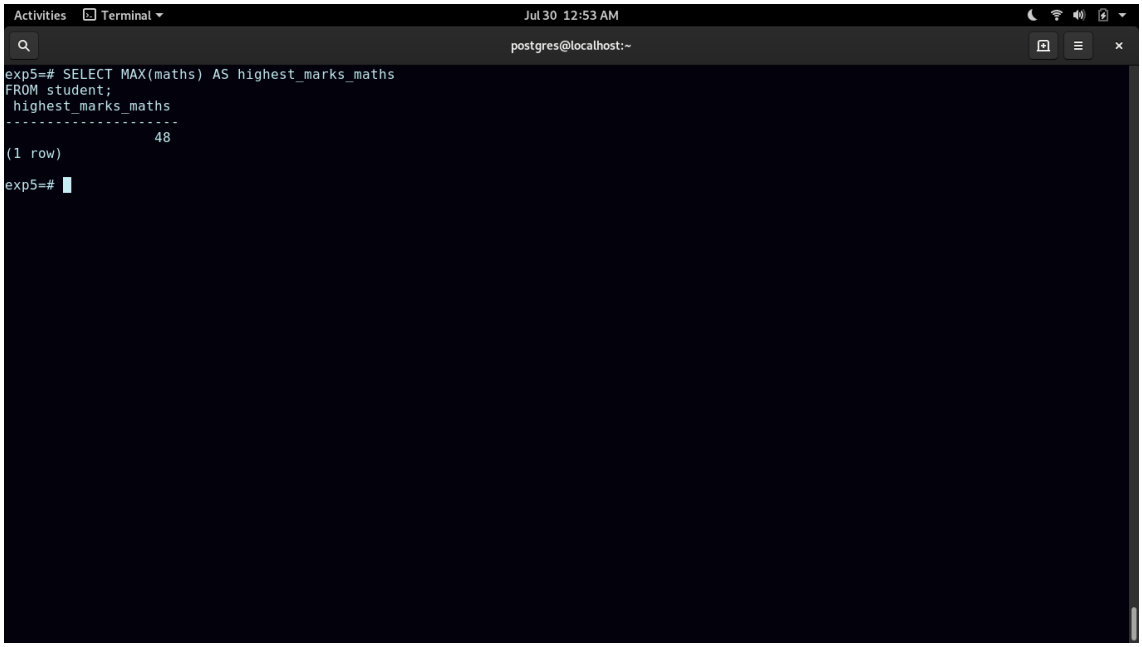

Figure 9: Question 2

3. Find the lowest marks for chemistry(To be displayed as lowest_mark_chemistry)

```
SELECT MIN(maths) AS
 lowest_marks_chemistry
FROM student;
```

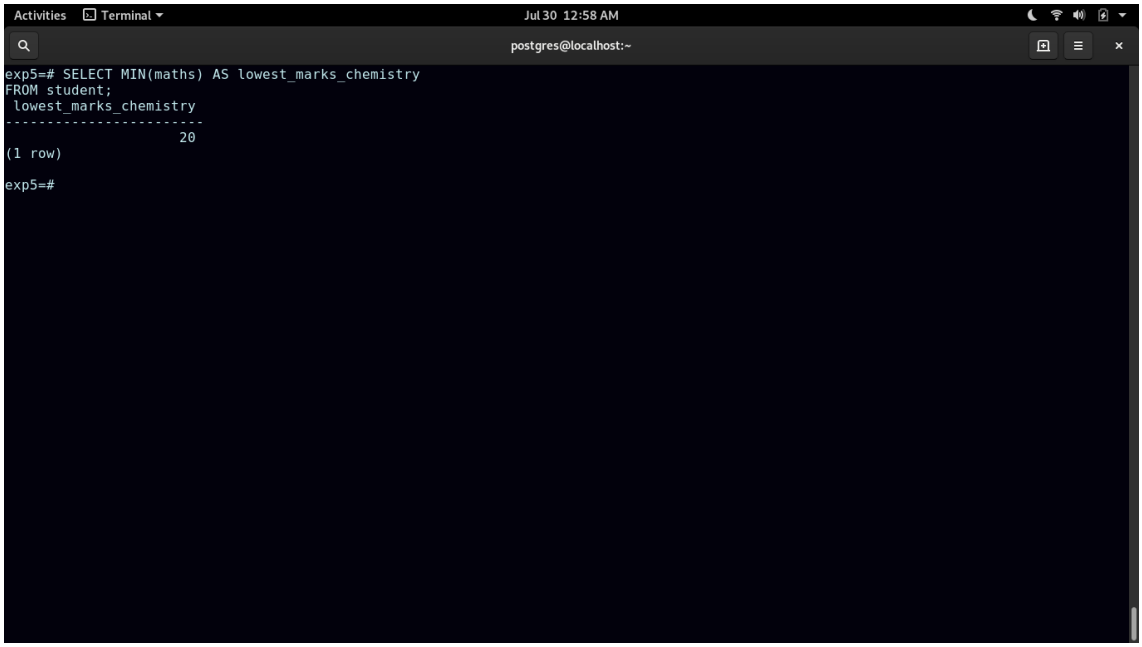

Figure 10: Question 3

4. Find the total number of students who has got a 'pass' in physics.

```
SELECT COUNT(*) AS count_rollno
FROM student
WHERE physics >= 12;
```

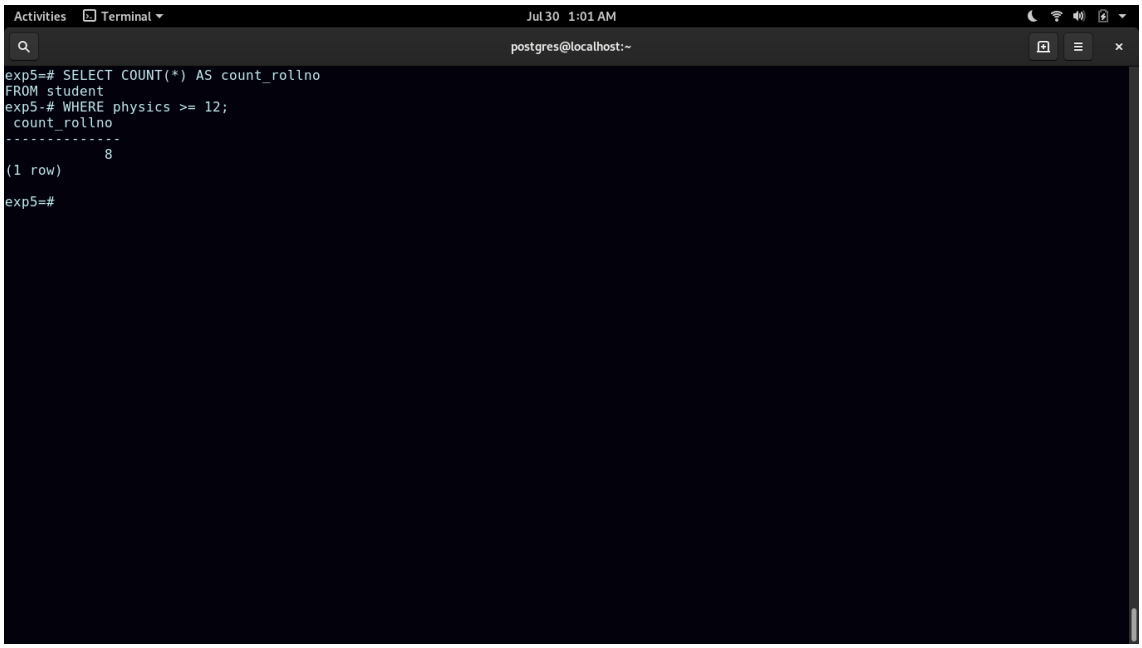

Figure 11: Question 4

5. Generate the list of students who have passed in all the subjects

```
SELECT * FROM student
WHERE physics >= 12
AND chemistry >= 12
AND maths >=25;
```

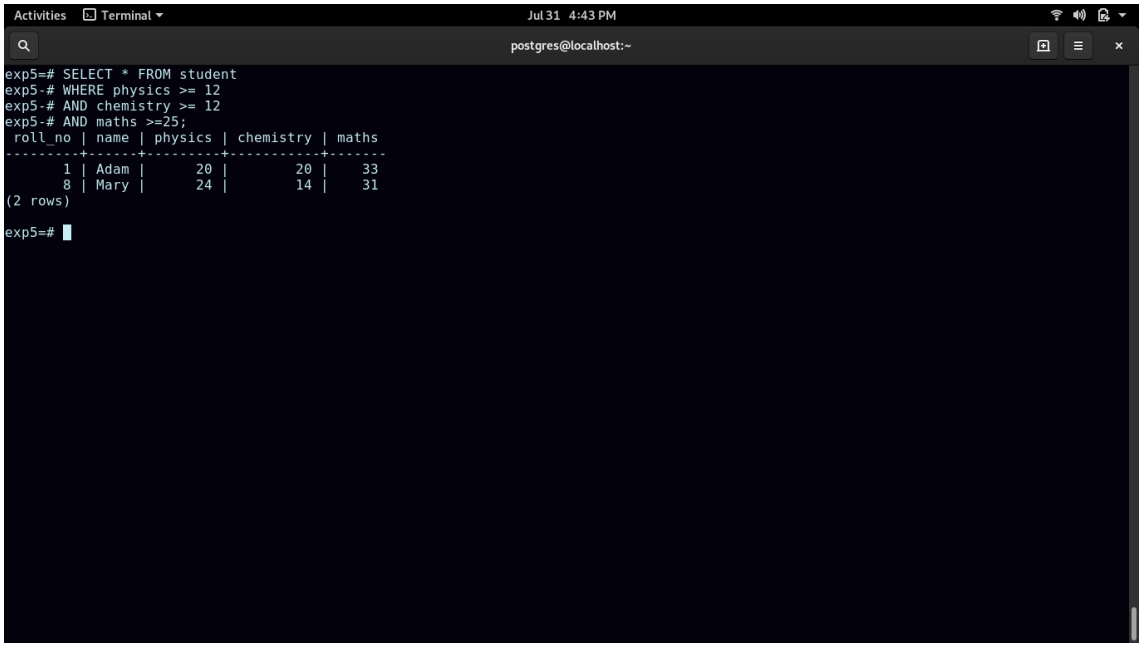

Figure 12: Question 5

6. Generate a rank list for the class.Indicate Pass/Fail. Ranking based on total marks obtained by the students.

```
SELECT *,
(physics+chemistry+maths) AS "Total Marks",
CASE
WHEN physics>=12 AND chemistry>=12 AND maths>=25
THEN 'P'
ELSE 'F'
END AS result
FROM student;
```

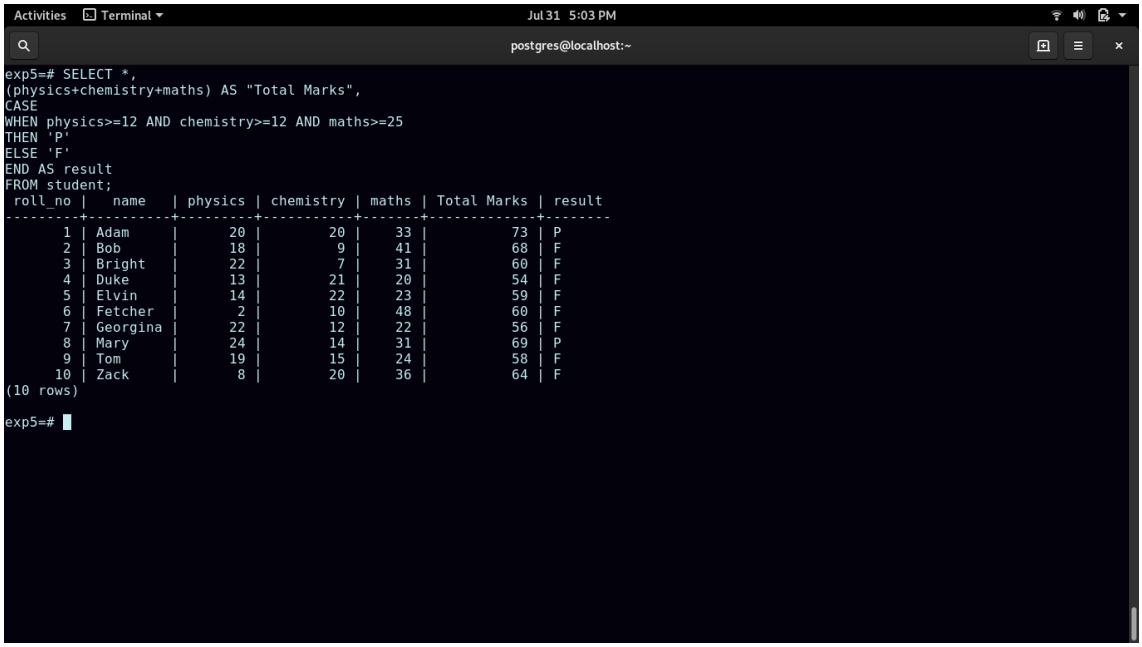

Figure 13: Question 6

7. Find pass percentage of the class for mathematics.

```
SELECT ((COUNT(maths) * 100 /
(SELECT COUNT(*) FROM student)))
AS pass_percentage_maths
FROM student
WHERE maths>=25;
```

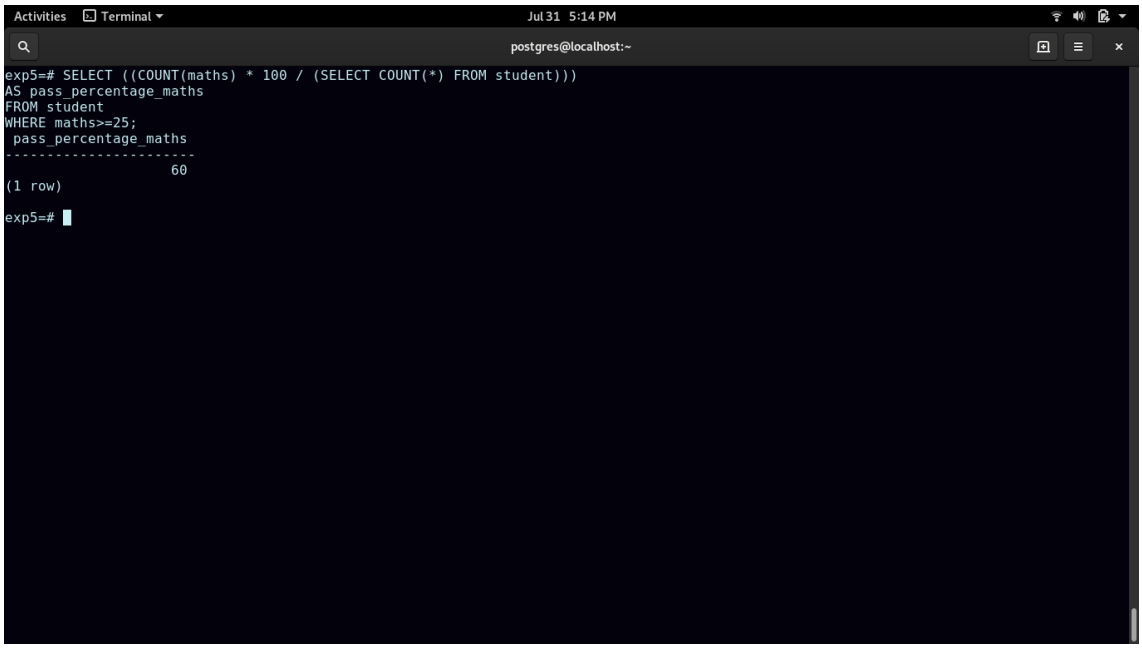

Figure 14: Question 7

8. Find the overall pass percentage for all class.

```
SELECT ((COUNT(*) * 100 /
(SELECT COUNT(*) FROM student)))
AS pass_percentage
FROM student
WHERE physics>=12 AND
chemistry>=12 AND maths>=25;
```

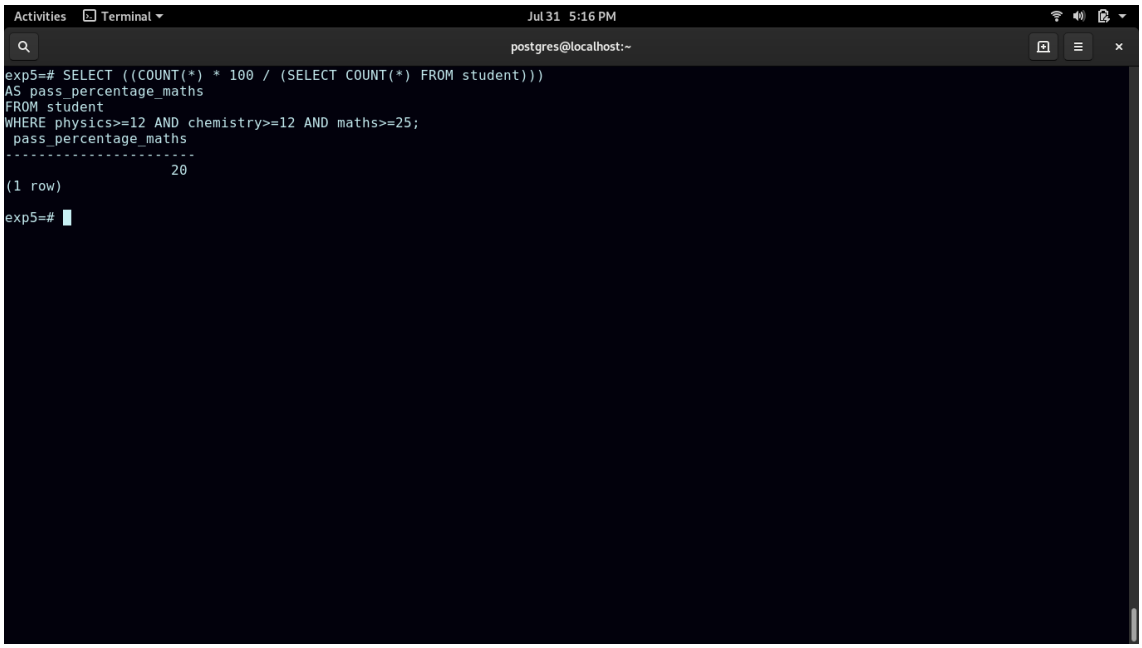

Figure 15: Question 8

9. Find the class average.

```
SELECT AVG(physics+chemistry+maths)
FROM student;
```

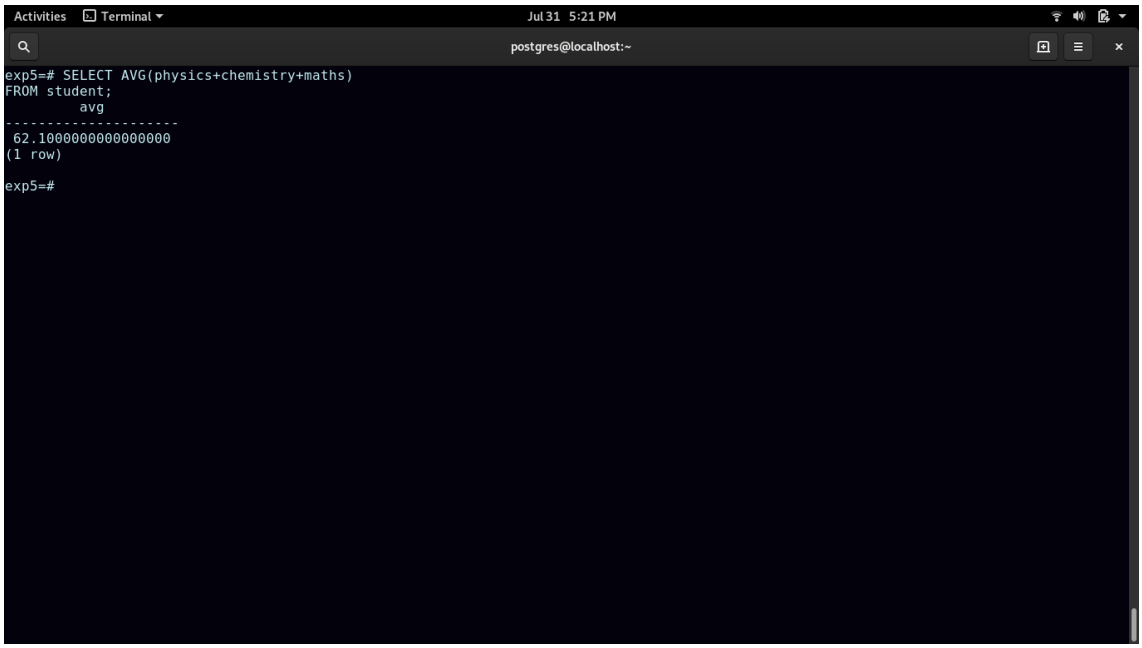

Figure 16: Question 9

10. Find the total number of students who have got a Pass.

```
SELECT COUNT(*) AS "Count(Result)"
FROM student
WHERE physics>=12 AND
 chemistry>=12 AND maths>=25;
```

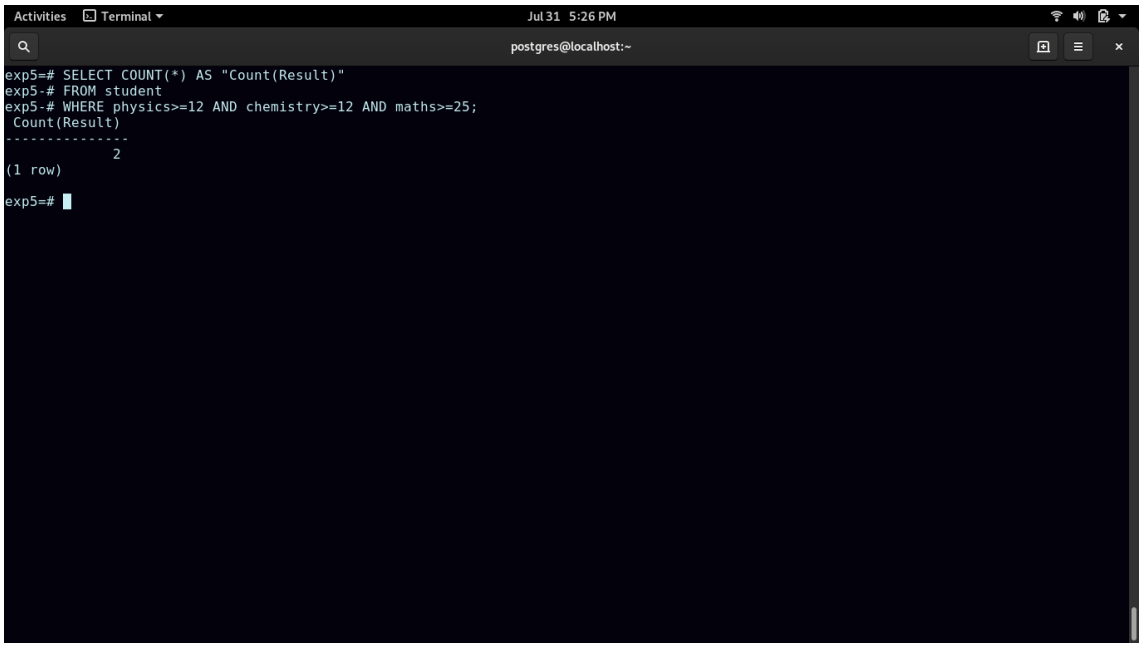

Figure 17: Question 10

4 Result

- Successfully used the aggregate functions AVG(), MAX(), MIN(), COUNT(), SUM() in PostgreSQL.