Exp No. 5 Date : 29-07-2019

AGGREGATE FUNCTIONS

AIM:

Introduction to Aggregate functions
- AVG ()
- MAX ()
- MIN ()
- COUNT ()
- SUM ()

Description:

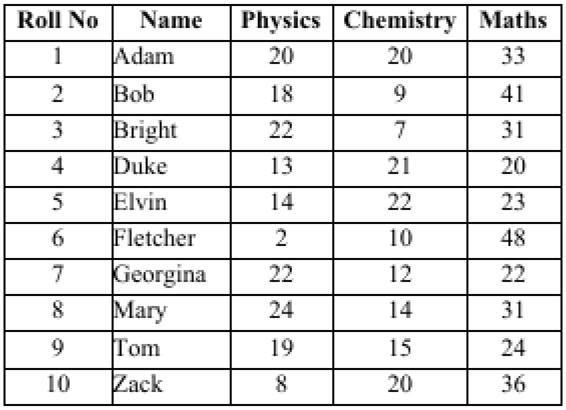
- Sum(field_name):
  Returns the total sum of the field.
- Avg(field_name):
  Returns the average of the field.
- Count( ) :
  Count function has three variations:
  i. Count(*) : Returns the number of rows in the table including duplicates and those with null values
  ii. Count(field_name) : returns the number of rows where field value is not null
  iii. Count (All): returns the total number of rows. It is same like count(*)

- Max(field_name):
  Returns the maximum value of the field
- Min(field_name):
  Returns the maximum value of the field

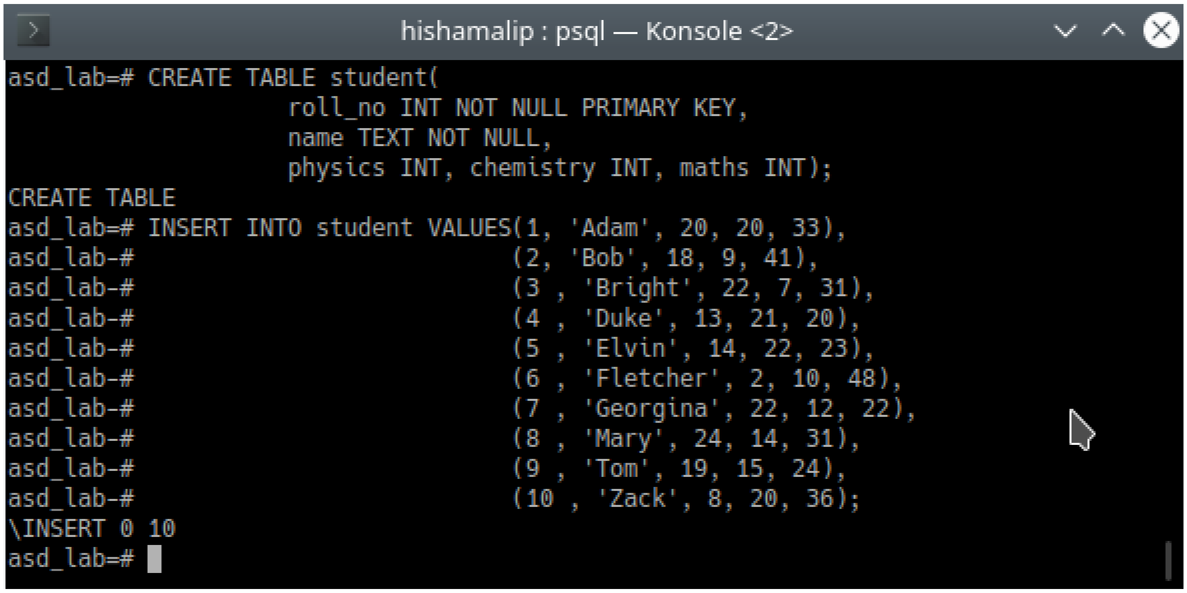
QUESTION:

Create a table named student and populate the table as shown in the table.

The table contains the marks of 10 students for 3 subjects(Physics, Chemistry, Mathematics).The total marks for physics and chemistry is 25.while for mathematics it is 50.The pass mark for physics and chemistry is 12 and for mathematics it is 25.A student is awarded a ‘Pass’ if he has passed all the subjects.

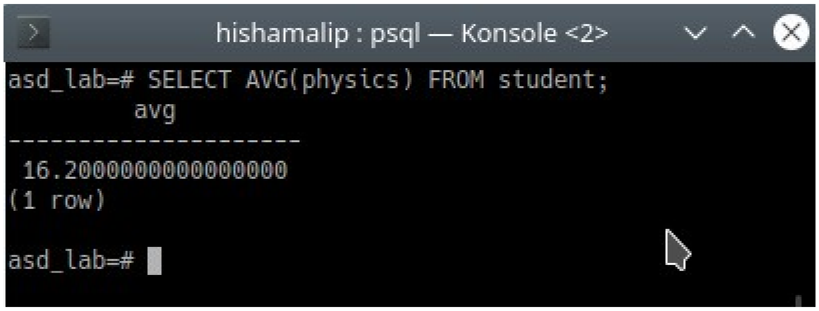


Roll No	Name	Physics	Chemistry	Maths
1	Adam	20	20	33
2	Bob	18	9	41
3	Bright	22	7	31
4	Duke	13	21	20
5	Elvin	14	22	23
6	Fletcher	2	10	48
7	Georgina	22	12	22
8	Mary	24	14	31
9	Tom	19	15	24
10	Zack	8	20	36



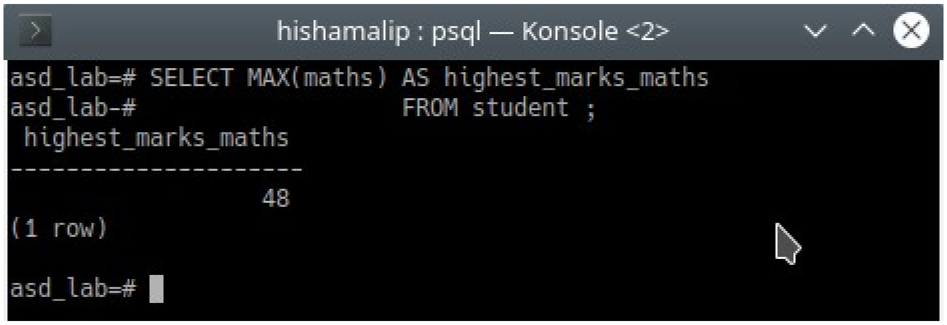
```
asd_lab=# CREATE TABLE student(
                    roll_no INT NOT NULL PRIMARY KEY,
                    name TEXT NOT NULL,
                    physics INT, chemistry INT, maths INT);
CREATE TABLE
asd_lab=# INSERT INTO student VALUES(1, 'Adam', 20, 20, 33),
asd_lab-#                           (2, 'Bob', 18, 9, 41),
asd_lab-#                           (3 , 'Bright', 22, 7, 31),
asd_lab-#                           (4 , 'Duke', 13, 21, 20),
asd_lab-#                           (5 , 'Elvin', 14, 22, 23),
asd_lab-#                           (6 , 'Fletcher', 2, 10, 48),
asd_lab-#                           (7 , 'Georgina', 22, 12, 22),
asd_lab-#                           (8 , 'Mary', 24, 14, 31),
asd_lab-#                           (9 , 'Tom', 19, 15, 24),
asd_lab-#                           (10 , 'Zack', 8, 20, 36);
\INSERT 0 10
asd_lab=# 
```

1. Find the class average for the subject ‘Physics’



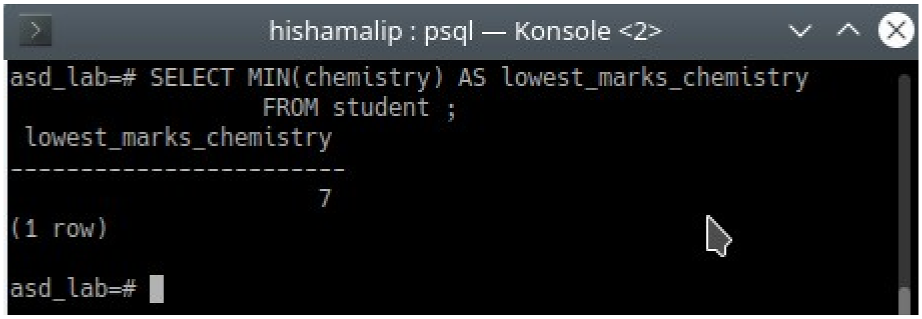
```
asd_lab=# SELECT AVG(physics) FROM student;
         avg
---------------------
 16.2000000000000000
(1 row)

asd_lab=# 
```

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



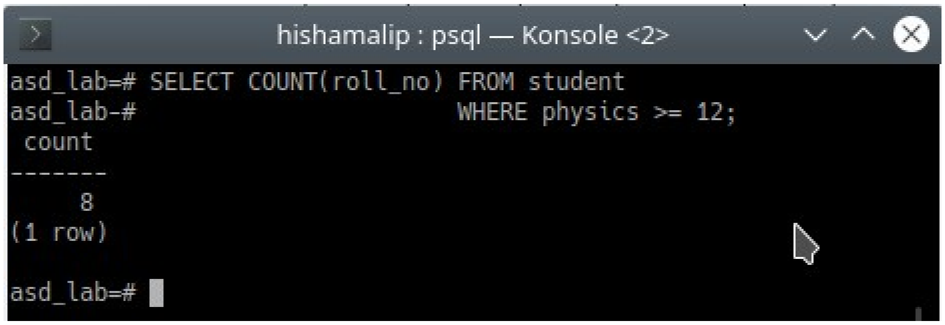
```
asd_lab=# SELECT MAX(maths) AS highest_marks_maths
asd_lab-#                   FROM student ;
 highest_marks_maths
---------------------
                  48
(1 row)

asd_lab=# 
```

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



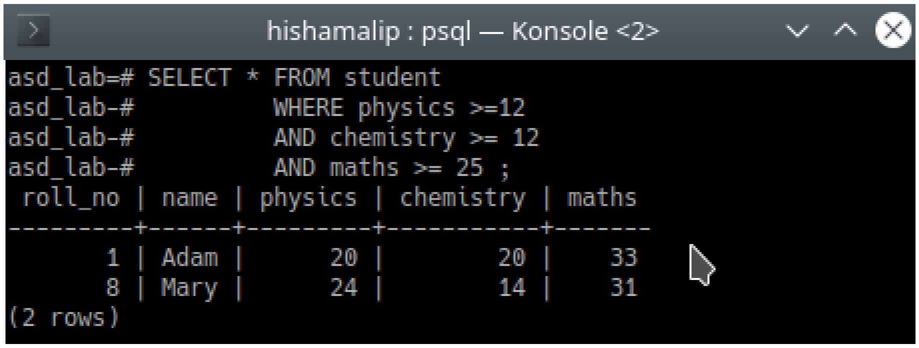
```
asd_lab=# SELECT MIN(chemistry) AS lowest_marks_chemistry
                  FROM student ;
 lowest_marks_chemistry
------------------------
                      7
(1 row)

asd_lab=# 
```

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



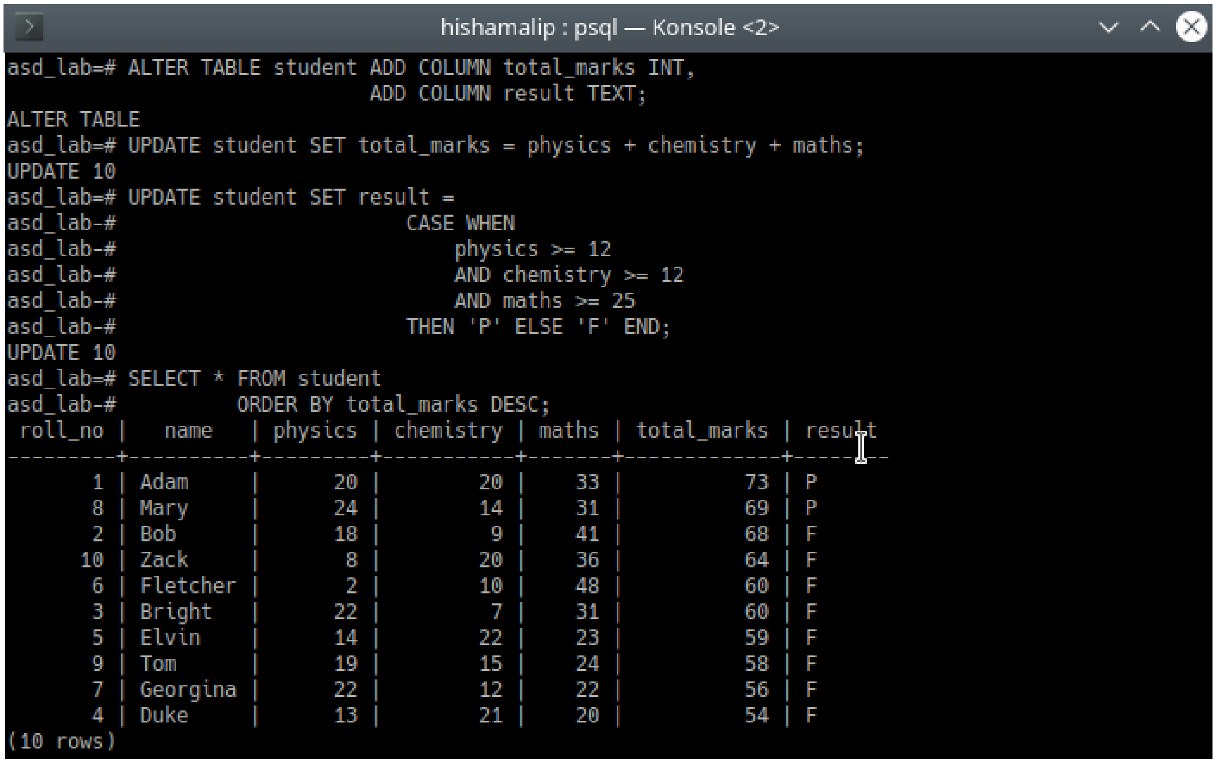
```
asd_lab=# SELECT COUNT(roll_no) FROM student
asd_lab-#                       WHERE physics >= 12;
 count
-------
     8
(1 row)

asd_lab=# 
```

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



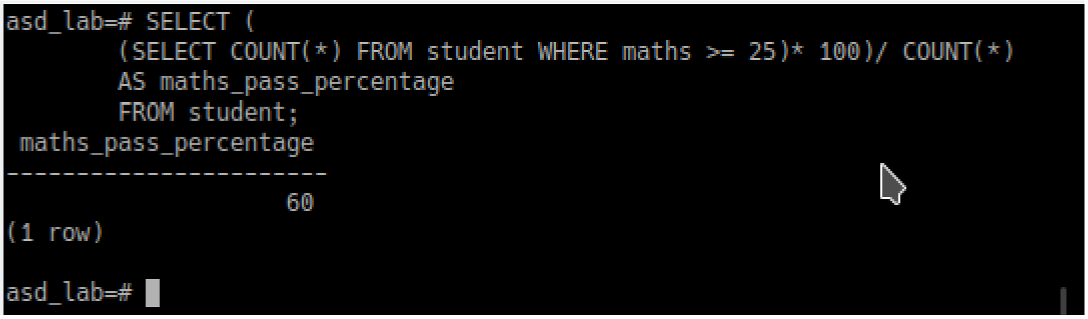
```
asd_lab=# SELECT * FROM student
asd_lab-#          WHERE physics >=12
asd_lab-#          AND chemistry >= 12
asd_lab-#          AND maths >= 25 ;
 roll_no | name | physics | chemistry | maths
---------+------+---------+-----------+-------
       1 | Adam |      20 |        20 |    33
       8 | Mary |      24 |        14 |    31
(2 rows)
```

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



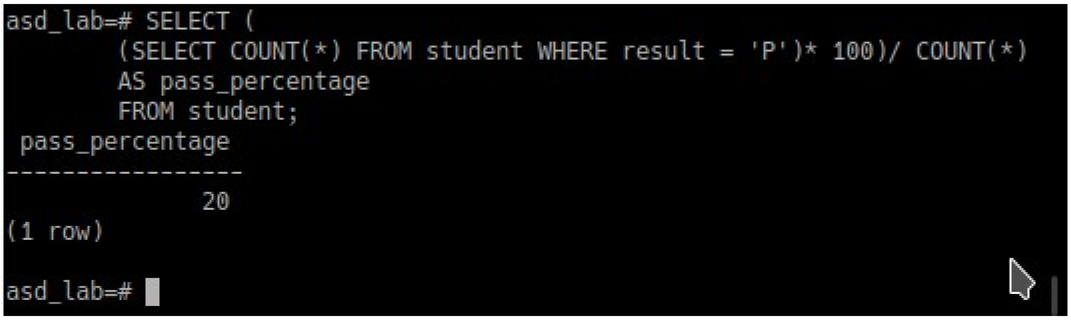
```
asd_lab=# ALTER TABLE student ADD COLUMN total_marks INT,
                              ADD COLUMN result TEXT;
ALTER TABLE
asd_lab=# UPDATE student SET total_marks = physics + chemistry + maths;
UPDATE 10
asd_lab=# UPDATE student SET result =
asd_lab-#                        CASE WHEN
asd_lab-#                            physics >= 12
asd_lab-#                            AND chemistry >= 12
asd_lab-#                            AND maths >= 25
asd_lab-#                        THEN 'P' ELSE 'F' END;
UPDATE 10
asd_lab=# SELECT * FROM student
asd_lab-#          ORDER BY total_marks DESC;
 roll_no |   name   | physics | chemistry | maths | total_marks | result
---------+----------+---------+-----------+-------+-------------+--------
       1 | Adam     |      20 |        20 |    33 |          73 | P
       8 | Mary     |      24 |        14 |    31 |          69 | P
       2 | Bob      |      18 |         9 |    41 |          68 | F
      10 | Zack     |       8 |        20 |    36 |          64 | F
       6 | Fletcher |       2 |        10 |    48 |          60 | F
       3 | Bright   |      22 |         7 |    31 |          60 | F
       5 | Elvin    |      14 |        22 |    23 |          59 | F
       9 | Tom      |      19 |        15 |    24 |          58 | F
       7 | Georgina |      22 |        12 |    22 |          56 | F
       4 | Duke     |      13 |        21 |    20 |          54 | F
(10 rows)
```

7. Find pass percentage of the class for mathematics.



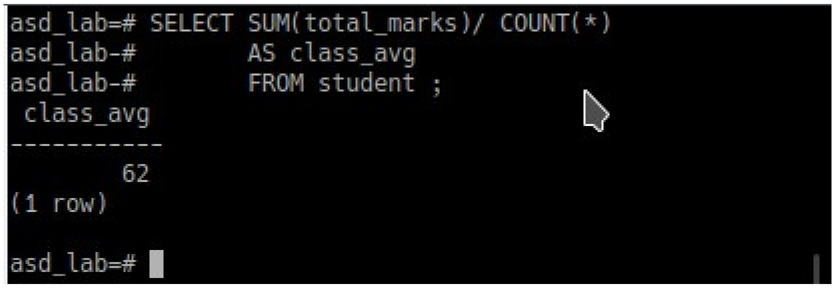
```
asd_lab=# SELECT (
        (SELECT COUNT(*) FROM student WHERE maths >= 25)* 100)/ COUNT(*)
        AS maths_pass_percentage
        FROM student;
 maths_pass_percentage
-----------------------
                    60
(1 row)

asd_lab=# 
```

8. Find the overall pass percentage for all class.



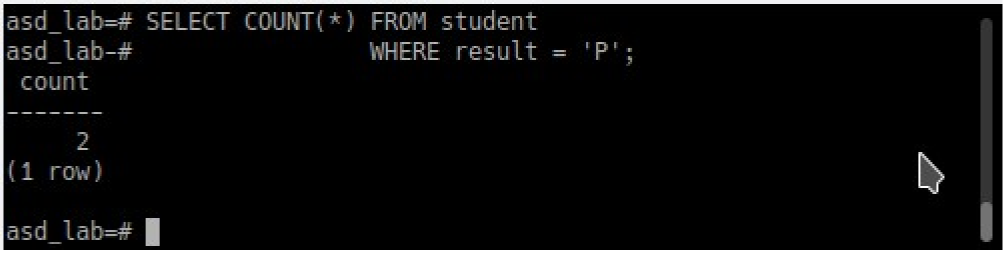
```
asd_lab=# SELECT (
        (SELECT COUNT(*) FROM student WHERE result = 'P')* 100)/ COUNT(*)
        AS pass_percentage
        FROM student;
 pass_percentage
-----------------
              20
(1 row)

asd_lab=# 
```

9. Find the class average.



```
asd_lab=# SELECT SUM(total_marks)/ COUNT(*)
asd_lab-#        AS class_avg
asd_lab-#        FROM student ;
 class_avg
-----------
        62
(1 row)

asd_lab=# 
```

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



```
asd_lab=# SELECT COUNT(*) FROM student
asd_lab-#                 WHERE result = 'P';
 count
-------
     2
(1 row)

asd_lab=# 
```

RESULT:

The query was executed successfully and output was obtained.