CYCLE 3

DATE : 05/09/2024

3.1 Question 1: Display the count of teachers teaching either two or three different subjects

```
SELECT count(staffid)
FROM Faculty_subjects
GROUP BY staffid
HAVING count(subjectid) = 2 OR count(subjectid) = 3;
```

OUTPUT :

```
s22b50=> select staffid,count(staffid) from faculty_subjects group by staffid having count(subjectid)=2 OR count(subjectid) = 3;
 staffid | count
---------+-------
      62 |     2
(1 row)
```

3.2 Question 2: Display the maximum of average salary from Staff table based on their positions

```
SELECT max(avg) FROM (
SELECT avg(salary) as avg
FROM Staff
GROUP BY position )
as max;
```

OUTPUT :

```
s22b50=> select position ,avg(salary) from staff
s22b50-> where stfstate= 'texas' OR stfstate='newyork' AND salary > 10000
s22b50-> group by position having avg(salary)>25000;
     position      |         avg
-------------------+--------------------
 head of deparment | 30000.000000000000
(1 row)
```

3.3 Question 3: Display the minimum grade of students for each subject from each class

```
SELECT max(s.grade) as minimum_grade, c.subjectid, c.classid
FROM Student_schedules AS s, Classes AS c
WHERE c.classid = s.classid
GROUP BY c.subjectid, c.classid;
```

OUTPUT :

```
s22b50=> select min(s.grade)as min_grade,c.subjectid,c.classid from student_schedules as s, classes as c
where c.classid = s.classid group by c.subjectid,c.classid;
 min_grade | subjectid | classid
-----------+-----------+---------
 C         |        59 |      75
 A         |        54 |      74
 A         |        56 |      78
 A         |        58 |      76
 B         |        57 |      79
(5 rows)
```

3.4 Question 4: Display subject name which contains character ‘a’

```
SELECT subjectname
FROM Subjects
WHERE subjectname LIKE ’%a%’;
```

OUTPUT :

```
s22b50=> select subjectname from subjects where subjectname like '%a%';
 subjectname
-------------
 maths
(1 row)
```

3.5 Question 5: Display the name of students having mobile no starting with ‘90’

```
SELECT studfirstname, studlastname
FROM Students
WHERE cast ( studphonenumber as text) LIKE ’90%’;
```

OUTPUT :

```
s22b50=> select stdfirstname,stdlastname from students where cast (stdphonenumber as text) like '90%';
 stdfirstname | stdlastname
--------------+-------------
 pain         | usumaki
 killer       | mikasa
(2 rows)
```

3.6 Question 6: Retrieve Class ID and maximum duration available in Classes where maximum duration is less than 2 hours

```
SELECT classid, max(duration)
FROM Classes
WHERE duration < 2
GROUP BY classid;
```

OUTPUT :

```
s22b50=> select classid,max(duration) from classes where duration < 2 group by classid;
 classid | max
---------+-----
      78 |   1
      76 |   1
      74 |   1
(3 rows)
```

3.7 Question 7:Display the list of students based on Student ID in ascending order and group by Subject Name

```
SELECT sb.subjectname, s.studentid, s.studfirstname, s.studlastname
FROM Students s, Student_schedules ss, Classes c, Subjects sb,
WHERE s.studentid = ss.studentid AND ss.classid = c.classid AND
c.subjectid = sb.subjectid
GROUP BY sb.subjectname, s.studentid
ORDER BY s.studentid;
```

OUTPUT

```
s22b50=>  select sb.subjectname,s.student_id,s.stdfirstname,s.stdlastname from students as s,student_schedules as ss,classes c,subjects as sb
s22b50-> where s.student_id = ss.studentid AND ss.classid=c.classid
s22b50-> AND c.subjectid=sb.subjectid group by sb.subjectname,s.student_id order by s.student_id;
 subjectname | student_id | stdfirstname | stdlastname
-------------+------------+--------------+-------------
 computer    |          1 | renjth       | rajan
 maths       |          2 | meamora      | deku
 chemistry   |          3 | naruto       | hatake
 physics     |          4 | pain         | usumaki
 english     |          5 | killer       | mikasa
(5 rows)
```

3.8 Question 8 : Display position and average salary of staff belonging to state “Texas” or “New York” where salary is more than 10000 and average salary is less than 25000

```
SELECT position, avg(salary)
FROM Staff
WHERE stfstate = ’Texas’ or stfstate = ’New York’ AND salary > 10000
GROUP BY position
HAVING avg(salary) > 25000;
```

OUTPUT :

```
s22b50=> select max(avg) from (select avg(salary) as avg from staff group by position) as max;
        max
--------------------
 30000.000000000000
(1 row)
```

RESULT

Various queries were executed in PostgreSQL and output was verified.

CYCLE 4

DATE : 05/09/2024

4.1 Question 1(a): Revoke insert privilege for a user on table Students and check whether you are able to insert a row in to the table

REVOKE INSERT ON STUDENTS FROM s22b50;

INSERT INTO STUDENTS VALUES (6, ’rreni’, ’paradil’, ’St. george Street’, ’dallas’, ‘196’,’ 2763’, 9188562613’);

OUTPUT :

```
s22b50=> REVOKE insert on students from s22b50;
REVOKE
s22b50=> insert into students values(6,'reni','padaril','st george street','dallas','196','2763','9188562613');
ERROR:  permission denied for relation students
```

4.2 Question 1(b): Grant the permission to the user for inserting values in to students table and check whether insertion is possible or not

GRANT INSERT ON STUDENTS TO s22b50;

INSERT INTO STUDENTS VALUES (6, ’rreni’, ’paradil’, ’St. george Street’, ’dallas’, ‘196’,’ 2763’, 9188562613’);

OUTPUT :

```
s22b50=> GRANT insert on students to  s22b50;
GRANT
s22b50=> insert into students values(6,'reni','padaril','st george street','dallas','196','2763','9188562613');
INSERT 0 1
s22b50=> select * from  students;
 student_id | stdfirstname | stdlastname |  stdstreetadress   |  stdcity   | stdzipcode | stdareacode | stdphonenumber
------------+--------------+-------------+--------------------+------------+------------+-------------+----------------
          1 | renjth       | rajan       | pottankad,achankad | adimaly    | 685565     | 685565      | 8078656135
          2 | meamora      | deku        | anamudi,estate     | munnar     | 685565     | 685578      | 8078651234
          3 | naruto       | hatake      | first street       | rajakumari | 685123     | 123456      | 8078651234
          4 | pain         | usumaki     | first street       | rajakumari | 686123     | 123456      | 9078651234
          5 | killer       | mikasa      | second street      | roman      | 686123     | 123456      | 9078651299
          6 | reni         | padaril     | st george street   | dallas     | 196        | 2763        | 9188562613
(6 rows)
```

4.3 Question 2(a): Start a new transaction and insert a row into the Staff table. Commit the transaction and display the changes to the table

BEGIN;

INSERT INTO STUDENTS VALUES (162, ’Jacob’, ’George’, ’St. Antony Street’, ’El Paso’, ’Texas’,

197, 2764, 9188562508);

COMMIT;

OUTPUT :

```
s22b50=> begin;
BEGIN
s22b50=> insert into staff values(50,'jenny','rose','jelgium street','ayon','texas','145','2298','9567890302','2002-04-29',11200,'professor');
INSERT 0 1
s22b50=> commit;
COMMIT
s22b50=> select * from  staff;
 staffid | stffirstname | stflastname | stfstreetadress  | stfcity | stfstate | stfzipcode | stfareacode | stfphonenumber | datehaired | salary |     position
---------+--------------+-------------+------------------+---------+----------+------------+-------------+----------------+------------+--------+------------------
      62 | oliva        | liam        | st thomas street | akron   | ohio     |        198 |        2245 | 9978564321     | 2002-03-12 |  11000 | ass.professor
      76 | kiran        | rick        | st jhons street  | akron   | ohio     |        198 |        2245 | 9978567788     | 2001-03-10 |  13000 | ass.professor
      12 | rose         | rick        | st don street    | oakland | newyork  |        198 |        2255 | 9922567488     | 1995-03-10 |  23000 | professor
      10 | ria          | havick      | st don street    | oakland | newyork  |        198 |        2255 | 9921122333     | 2000-07-24 |  30000 | head of deparment
      45 | morty        | mick        | shepard  street  | bufate  | texas    |        145 |        2295 | 9821222887     | 2008-08-14 |  23000 | ass.professor
      50 | jenny        | rose        | jelgium street   | ayon    | texas    |        145 |        2298 | 9567890302     | 2002-04-29 |  11200 | professor
(6 rows)
```

4.4 Question 2(b): Start a new transaction and insert a row into the Staff table.Undo the transaction and display the changes to the table

BEGIN;
INSERT INTO STUDENTS VALUES (163, ’John’, ’George’, ’St. Peter Street’, Buffalo’, ’New
York’, 196, 2763, 9188562507);
ROLLBACK;

OUTPUT :

```
BEGIN
s22b50=> insert into students values(7,'rahul','raju','second street','dallas','197','2793','1212121212');
INSERT 0 1
s22b50=> rollback;
ROLLBACK
s22b50=> select * from  students;
 student_id | stdfirstname | stdlastname |  stdstreetadress   |  stdcity   | stdzipcode | stdareacode | stdphonenumber
------------+--------------+-------------+--------------------+------------+------------+-------------+----------------
          1 | renjth       | rajan       | pottankad,achankad | adimaly    | 685565     | 685565      | 8078656135
          2 | meamora      | deku        | anamudi,estate     | munnar     | 685565     | 685578      | 8078651234
          3 | naruto       | hatake      | first street       | rajakumari | 685123     | 123456      | 8078651234
          4 | pain         | usumaki     | first street       | rajakumari | 686123     | 123456      | 9078651234
          5 | killer       | mikasa      | second street      | roman      | 686123     | 123456      | 9078651299
          6 | reni         | padaril     | st george street   | dallas     | 196        | 2763        | 9188562613
(6 rows)
```

4.5 Question 3: Display the staffid and title for Faculty along with staffid and position for Staff in a single table. Indicate the source of the row in the result by adding an additional column EMPLOYEE with possible values as ‘F’ (Faculties) and ‘S’ (Staff). Display all rows (Using UNION ALL)

SELECT staffid, title, ’F’ as Employee
FROM Faculty
UNION ALL
SELECT staffid, position, ’S’ as Employee
FROM Staff

OUTPUT :

```
s22b50=> select staffid,title,'F' as employee
s22b50-> from faculty
s22b50-> union all
s22b50-> select staffid,position ,'S' as emlpoyee
s22b50-> from staff;
 staffid |       title       | employee
---------+-------------------+----------
      76 | ass.professor     | F
      12 | professor         | F
      10 | head of deparment | F
      45 | ass.professor     | F
      62 | ass.professor     | F
      62 | ass.professor     | S
      76 | ass.professor     | S
      12 | professor         | S
      10 | head of deparment | S
      45 | ass.professor     | S
      50 | professor         | S
(11 rows)
```

4.6 Question 4: Find the pass percentage of a particular subject (using grade)

SELECT s.subjectname, 100.0*COUNT(*)::DECIMAL/(SELECT COUNT(*)
FROM Subjects s, Classes c, Student_schedules ss
WHERE s.subjectid=c.subjectid AND c.classid=ss.classid AND
s.subjectname='Physics')
AS pass_percentage
FROM Subjects s, Classes c, Student_schedules ss
WHERE s.subjectid = c.subjectid AND c.classid=ss.classid AND
s.subjectname='Physics'
AND ss.grade < 'E' GROUP BY s.subjectname;

OUTPUT :

```
SELECT s.subjectname, 100.0*COUNT(*)::DECIMAL/(SELECT COUNT(*)
FROM Subjects s, Classes c, Student_schedules ss
WHERE s.subjectid=c.subjectid AND c.classid=ss.classid AND s.subjectname='chemistry')
AS pass_percentage
FROM Subjects s, Classes c, Student_schedules ss
WHERE s.subjectid = c.subjectid AND c.classid=ss.classid AND s.subjectname='chemistry'
AND ss.grade < 'C' GROUP BY s.subjectname;
 subjectname |   pass_percentage
-------------+---------------------
 chemistry   | 100.0000000000000000
(1 row)
```

4.7 Question 5: Display the number of students in each classroom on a

particular building using JOINS

```
SELECT buildingcode, buildingname, classid, count(studentid) as
number_of_students
FROM Student_schedules NATURAL JOIN Classes NATURAL JOIN
Classrooms NATURAL JOIN
Buildings
GROUP BY buildingcode, classid, buildingname;
```

OUTPUT :

```
s22b50=> SELECT buildcode, buildingname, classid, count(studentid) as number_of_students
FROM Student_schedules NATURAL JOIN Classes NATURAL JOIN Class_rooms NATURAL JOIN
Building
GROUP BY buildcode, classid, buildingname;
 buildcode | buildingname | classid | number_of_students
-----------+--------------+---------+--------------------
 cse       | cse dept     |      79 |                  1
 ec        | ec dept      |      74 |                  1
 cse       | cse dept     |      78 |                  1
 me        | mech dept    |      76 |                  1
 ee        | electrical   |      75 |                  1
(5 rows)
```

4.8 Question 6: Display the list of students and staff who have the same zip Code

```
SELECT stffirstname, studfirstname, studzipcode as zipcode
FROM Staff JOIN Students
ON Staff.zipcode = Students.zipcode;
```

OUTPUT :

```
s22b50=> select stdfirstname,stffirstname,stfzipcode as zipcode from staff JOIN students ON cast (staff.stfzipcode as text)=students.stdzipcode;
 stdfirstname | stffirstname | zipcode
--------------+--------------+---------
 reni         | jenny        |     145
 reni         | morty        |     145
(2 rows)
```

4.9 Question 7: Display the list of faculty who engage same subject (for any particular subject name)

```
SELECT fs.subjectid, sub.subjectname, sf.staffid, sf.stffirstname
FROM Subjects sub, Staff sf, Faculty_subjects fs
WHERE fs.staffid=sf.staffid AND fs.subjectid=sub.subjectid AND
sub.subjectname='Chemistry';
```

OUTPUT :

```
s22b50=> select fs.subjectid,sub.subjectname,sf.staffid,sf.stffirstname
from subjects sub,faculty_subjects fs,staff sf
where fs.staffid=sf.staffid AND sub.subjectid=fs.subjectid AND sub.subjectname='chemistry';
 subjectid | subjectname | staffid | stffirstname
-----------+-------------+---------+--------------
        58 | chemistry   |      12 | rose
        58 | chemistry   |      45 | morty
(2 rows)
```

RESULT

Various queries were executed in PostgreSQL and output was verified.

CYCLE 5