ASSIGNMENT-1

1.For each Computer Sciences class, print the cno, sectno, and the average gpa of the students enrolled in the class. (6 pts)

Query:

```
SELECT c.cno as course_number, e.sectno as section_number, AVG(e.grade) as avg_gpa
FROM course c
JOIN enroll e ON c.cno = e.cno
JOIN student st ON st.sid = e.sid
WHERE c.dname = 'Computer Sciences'
GROUP BY c.cno, e.sectno;
```

```
skunamne=> SELECT c.cno as course_number, e.sectno as section_number, AVG(e.grade) as avg_gpa FROM course c
JOIN enroll e ON c.cno = e.cno
JOIN student st ON st.sid = e.sid
WHERE c.dname = 'Computer Sciences'
GROUP BY c.cno, e.sectno;
 course_number | section_number |      avg_gpa
---------------+----------------+--------------------
           302 |              1 | 2.9500000000000000
           302 |              2 | 3.1000000000000000
           467 |              1 | 3.1000000000000000
           701 |              1 | 3.2083333333333333
           726 |              1 | 3.2352941176470588
(5 rows)
```

2.Print the course names, course numbers and section numbers of all classes with less than six students enrolled in them. (8 pts)

Query:

```
SELECT c.cname, c.cno, e.sectno,COUNT(e.sid) as count
FROM course c
LEFT JOIN enroll e on e.cno = c.cno
GROUP BY c.cname, c.cno, e.sectno
HAVING COUNT(e.sid) < 6;
```

```
skunamne=> SELECT c.cname, c.cno, e.sectno,COUNT(e.sid) as count
FROM course c
LEFT JOIN enroll e on e.cno = c.cno
GROUP BY c.cname, c.cno, e.sectno
HAVING COUNT(e.sid) < 6;
 cname | cno | sectno | count
-------+-----+--------+-------
(0 rows)
```

3.Print the names of departments that have one or more majors who are under 18 years old. (8 pts)

Query:

SELECT dept.dname as dept_name FROM dept

INNER JOIN major ON dept.dname = major.dname

INNER JOIN student ON major.sid = student.sid

WHERE age < 18;

```
skunamne=> SELECT dept.dname as dept_name FROM dept
INNER JOIN major ON dept.dname = major.dname
INNER JOIN student ON major.sid = student.sid
WHERE age < 18;
       dept_name
------------------------
 Industrial Engineering
 Mathematics
(2 rows)
```

4.Print the names and majors of students who are taking one of the College Geometry courses. For students who do not have a major, print out some special string, e.g., an empty string, as his/her major. (Hint: You'll need to use the "like" predicate and the string matching character in your query. You also likely need to use SET operations.) (12 pts)

Query:

WITH students_major AS (

SELECT st.sid, st.sname, m.dname as major

FROM student st

LEFT JOIN major m ON st.sid = m.sid

)

SELECT sm.sid as student_id, sm.sname as student_name, COALESCE(sm.major, '') as major

FROM students_major sm

JOIN enroll e ON sm.sid = e.sid

JOIN course c ON e.cno = c.cno AND e.dname = c.dname

WHERE c.cname LIKE '%College Geometry%';

```
skunamne=> WITH students_major AS (
SELECT st.sid, st.sname, m.dname as major
FROM student st
LEFT JOIN major m ON st.sid = m.sid
)
SELECT sm.sid as student_id, sm.sname as student_name, COALESCE(sm.major, '') as major
FROM students_major sm
JOIN enroll e ON sm.sid = e.sid
JOIN course c ON e.cno = c.cno AND e.dname = c.dname
WHERE c.cname LIKE '%College Geometry%';
 student_id |   student_name    |        major
------------+-------------------+---------------------
          4 | Sulfate, Barry M. | Computer Sciences
          4 | Sulfate, Barry M. | Sanitary Engineering
         14 | Cheong, R.        | Computer Sciences
         17 | Thorton, James Q. | Computer Sciences
         18 | Gooch             | Computer Sciences
         19 | Smith, L.         | Computer Sciences
         26 | Ford, Gerald      | Chemical Engineering
         28 | Austin, G.        | Chemical Engineering
         35 | Mathews, John W.  | Chemical Engineering
         37 | Dunbar, D.        | Civil Engineering
         40 | Rosemeyer, S.     | Civil Engineering
         53 | Atny, Mary H.     | Civil Engineering
         55 | Glitch, R.        | Civil Engineering
         59 | Ziebart, F.       | Civil Engineering
         90 | Zappa, F.         | Mathematics
         91 | Ghandi, I.        | Mathematics
         94 | Uoiea, Z.         | Mathematics
        101 | Davis, Scott P.   |
        102 | Bates, Michael L. |
(19 rows)
```

5.For those departments that have no majors taking a College Geometry course, print the department name and the number of PhD students in the department. (12 pts)

Query:

SELECT d.dname as dept_name, d.numphds as no_of_phd_students

FROM dept d

LEFT JOIN course c ON c.dname = d.dname

AND c.cname LIKE '%College Geometry%'

WHERE c.dname is NULL;

```
skunamne=> SELECT d.dname as dept_name, d.numphds as no_of_phd_students
FROM dept d
LEFT JOIN course c ON c.dname = d.dname
AND c.cname LIKE '%College Geometry%'
WHERE c.dname is NULL;
       dept_name        | no_of_phd_students
------------------------+--------------------
 Chemical Engineering   |                 32
 Civil Engineering      |                 88
 Computer Sciences      |                 47
 Industrial Engineering |                 41
 Sanitary Engineering   |                  3
(5 rows)
```

6. Print the names of students who are taking both a Computer Sciences course and a Mathematics course. (12 pts)

Query:

SELECT st.sid as student_id, st.sname as student_name

FROM student st

JOIN enroll en1 ON en1.sid=st.sid JOIN course c1 ON c1.cno=en1.cno

JOIN enroll en2 ON en2.sid=st.sid JOIN course c2 on c2.cno=en2.cno

WHERE c1.dname LIKE '%Mathematics%'

AND c2.dname LIKE '%Computer Science%'

GROUP BY st.sid, st.sname;

```
skunamne=> SELECT st.sid as student_id, st.sname as student_name
FROM student st
JOIN enroll en1 ON en1.sid=st.sid JOIN course c1 ON c1.cno=en1.cno
JOIN enroll en2 ON en2.sid=st.sid JOIN course c2 on c2.cno=en2.cno
WHERE c1.dname LIKE '%Mathematics%'
AND c2.dname LIKE '%Computer Science%'
GROUP BY st.sid, st.sname;
 student_id | student_name
------------+--------------
         90 | Zappa, F.
(1 row)
```

7. Print the age difference between the oldest and youngest Computer Sciences major(s). (10 pts)

Query:

```
SELECT MAX(age) - MIN(age) as age_difference
FROM student st
JOIN major m ON st.sid = m.sid
JOIN dept d ON m.dname = d.dname
WHERE d.dname = 'Computer Sciences';
```

```
skunamne=> SELECT MAX(age) - MIN(age) as age_difference
FROM student st
JOIN major m ON st.sid = m.sid
JOIN dept d ON m.dname = d.dname
WHERE d.dname = 'Computer Sciences';
 age_difference
----------------
             38
(1 row)
```

8.For each department that has one or more majors with a GPA under 1.0, print the name of the department and the average GPA of its majors. (15 pts)

Query:

```
select m.dname as dept_name, avg(st.gpa) as avg_gpa from major m
inner join student st
on st.sid = m.sid
group by m.dname
having min(st.gpa) < 1.0;
```

```
skunamne=> select m.dname as dept_name, avg(st.gpa) as avg_gpa from major m
inner join student st
on st.sid = m.sid
group by m.dname
having min(st.gpa) < 1.0;
       dept_name        |      avg_gpa
------------------------+--------------------
 Civil Engineering      | 2.9142857142857143
 Industrial Engineering | 2.7700000000000000
 Computer Sciences      | 3.0041666666666667
(3 rows)
```

9. Print the ids, names, and GPAs of the students who are currently taking all of the Civil Engineering courses. (15 pts)

Query:

SELECT st.sid, st.sname, st.gpa

FROM student st

JOIN (

SELECT e.sid

FROM enroll e

JOIN course c on e.cno=c.cno AND e.dname=c.dname

WHERE c.dname='Civil Engineering'

GROUP BY e.sid

HAVING COUNT(DISTINCT e.cno)=(SELECT count(*) from course where dname = 'Civil Engineering')

) t ON st.sid=t.sid;

```
skunamne=> SELECT st.sid, st.sname, st.gpa
FROM student st
JOIN (
SELECT e.sid
FROM enroll e
JOIN course c on e.cno=c.cno AND e.dname=c.dname
WHERE c.dname='Civil Engineering'
GROUP BY e.sid
HAVING COUNT(DISTINCT e.cno)=(SELECT count(*) from course where dname = 'Civil Engineering')
) t ON st.sid=t.sid;
 sid |    sname     | gpa
-----+--------------+------
  29 | Hamilton, S. | 2.80
(1 row)
```

PART-B

a. Find the average price (one average price) of PC's and laptops (combined) made by manufacturer "D."

Query:

select avg(R.price) as average from

((select pc.price from pc, product where pc.model = product.model and product.maker = 'D')

UNION

(select laptop.price from laptop, product where laptop.model = product.model and product.maker = 'D')) as R;

```
skunamne=> select avg( R.price) as average  from
(( select pc.price  from pc, product  where pc.model = product.model and product.maker = 'D' )
UNION
( select laptop.price from laptop, product where laptop.model = product.model and product.maker = 'D')) as R;
      average
--------------------
 25000.000000000000
(1 row)
```

b. Find the average hard disk size of a PC for all those manufacturers that make printers.

Query:

select pr. maker, avg(hd) from product pr,

(select maker from product pr where pr.type = 'Printer' group by maker) as result, pc

where pr.maker = result.maker and pr.model = pc.model

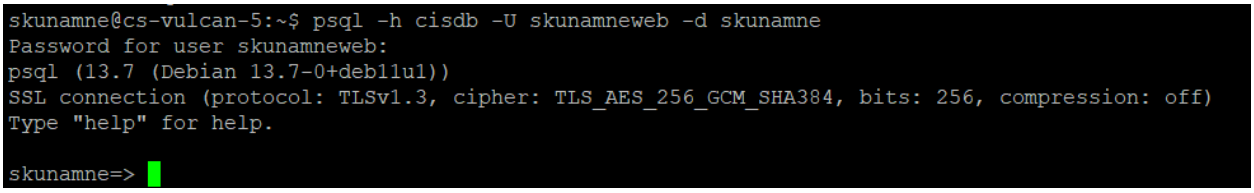
group by pr.maker;

```
skunamne=> select pr. maker , avg(hd) from product pr,
(select maker from product pr where pr.type = 'Printer' group by maker) as result, pc
where pr.maker = result.maker and pr.model = pc.model
group by pr.maker;
 maker |         avg
-------+----------------------
 D     | 875.0000000000000000
 A     | 500.0000000000000000
 C     | 625.0000000000000000
 B     | 583.3333333333333333
(4 rows)
```

ADDITIONAL INFORMATION:

Username: skunamne

Password:vidya1311



Database Tables:

Part A:

course, dept, enroll, major, prof, section, student

Part B:

product, laptop , pc, printer