COMP 533 HW2

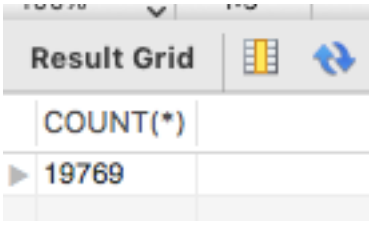
Mo Tang mt60 S01276578

1.

(a)

```
SELECT COUNT(*) From(
SELECT  name, email, phone  FROM central_contacts AS c1
UNION ALL
SELECT name,  email, phone FROM result_contacts AS c2
UNION ALL
SELECT  name, email, phone FROM central_contacts AS c3)
As total;
```

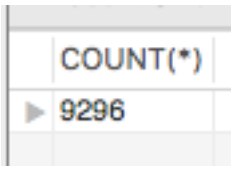
Count: 19769



(b)

```
SELECT COUNT(*) From(
SELECT  name, email, phone  FROM central_contacts AS c1
UNION
SELECT name,  email, phone FROM result_contacts AS c2
UNION
SELECT  name, email, phone FROM central_contacts AS c3)
As total;
```

Count: 9296



(c)

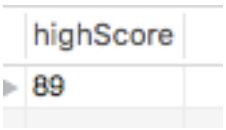
In current database, the same person may have different name. Like “MD” with “M.D.” We could put the name in a more standard way, like every name can only have one abbreviation, and divide the whole name to Job Title, First Name, Last Name.

2.

(a) The value has every condition.

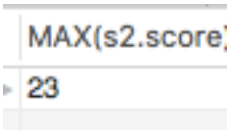
```
SELECT SUM(p.points) AS highScore FROM scorePoints AS p WHERE p.points>0;
```

Value: 89



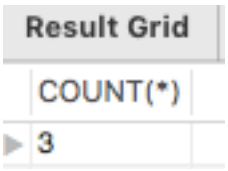
highScore
89

(b)
SELECT MAX(s2.score) FROM
(SELECT nct_id, SUM(sp.points) AS score
FROM
(SELECT s.nct_id, t.term
FROM studies AS s
INNER JOIN conditions AS c ON c.nct_id = s.nct_id INNER JOIN scoreTerms AS t ON t.name = c.name GROUP BY s.nct_id, t.term)
AS t
INNER JOIN scorePoints AS sp ON sp.term = t.term GROUP BY t.nct_id)
AS s2;
Score value : 23



MAX(s2.score)
23

(c)
SELECT COUNT(*)
FROM(
SELECT nct_id, SUM(sp.points) AS score
FROM(
SELECT s.nct_id, t.term
FROM studies AS s
INNER JOIN conditions AS co ON co.nct_id = s.nct_id
INNER JOIN scoreTerms AS t ON t.name = co.name AND t.term = "Neurodegenerative disorders"
GROUP BY s.nct_id, t.term)
AS t
INNER JOIN scorePoints AS sp ON sp.term = t.term GROUP BY t.nct_id) study_score WHERE score = 6;

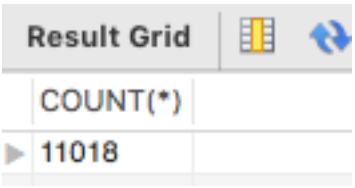


Result Grid

COUNT(*)
3

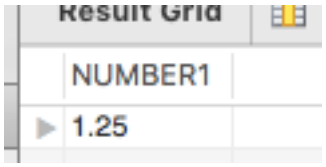
(d) studies: 11018
SELECT COUNT(*)
FROM studies AS s
WHERE s.nct_id NOT IN
(SELECT nct_id FROM
(SELECT DISTINCT s.nct_id
FROM studies AS s

```
INNER JOIN
(SELECT *
FROM conditions AS co
WHERE co.name IN (SELECT name FROM scoreTerms))
AS
r ON r.nct_id = s.nct_id) AS risk);
```



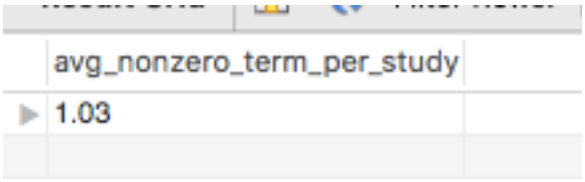
(e)

```
SELECT round(avg(num),2) AS NUMBER1 FROM (
SELECT nct_id, COUNT(i) AS num
FROM
(SELECT s.nct_id, co.id AS i
FROM studies AS s
INNER JOIN conditions AS co ON co.nct_id = s.nct_id
INNER JOIN scoreTerms AS st ON st.name = co.name
GROUP BY s.nct_id, co.id)
AS ns
GROUP BY nct_id) AS non;
```




```
SELECT ROUND(AVG(num),2) AS NUMBER2 FROM (
SELECT nct_id, count(term) AS num
FROM
(SELECT nct_id, term
FROM
(SELECT s.nct_id, st.term
FROM studies AS s
INNER JOIN conditions AS co ON co.nct_id = s.nct_id INNER JOIN scoreTerms AS st ON st.name
= co.name GROUP BY s.nct_id, st.term)
AS st
WHERE st.nct_id IN ( SELECT nct_id FROM
(SELECT nct_id, SUM(sp.points) AS score
FROM
(SELECT s.nct_id, st.term
FROM studies AS s
INNER JOIN conditions AS co ON co.nct_id = s.nct_id INNER JOIN scoreTerms AS st ON st.name
= co.name GROUP BY s.nct_id, st.term)
AS st
```

```
INNER JOIN scorePoints AS sp ON sp.term = st.term GROUP BY st.nct_id)
AS sco WHERE score !=0))
AS nn
GROUP BY nct_id) AS non;
```



avg_nonzero_term_per_study
1.03

3.

The current tables have : “term-points”, we just need add a new table:”term-newPoints”. The identical data structure will make all the operation the same.

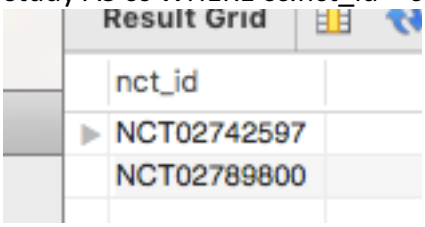
4.

The terms with negative points influence the final score.

5.

```
CREATE VIEW study AS
SELECT DISTINCT st.nct_id, co.name AS na FROM studies AS st
INNER JOIN conditions AS co ON st.nct_id = co.nct_id GROUP BY st.nct_id, co.name;

SELECT DISTINCT nct_id
FROM study AS ca WHERE NOT EXISTS
(SELECT cb.na FROM study AS cb WHERE cb.nct_id = "NCT02789800" AND
cb.na NOT IN (SELECT na FROM study AS cc WHERE cc.nct_id = ca.nct_id) );
```



Result Grid

nct_id
NCT02742597
NCT02789800

6.

(a)

0.0204; NCT02742597-NCt02595866 , NCT02742597-NCT03002311

```
CREATE VIEW jindex AS
SELECT j.id1, j.id2, ROUND(( j.common/(j.val_a+j.val_b - j.common)),4) AS ja FROM
(SELECT ic.id1, ic.id2, ic.inter_cnt AS common, gc1.cnt AS val_a, gc2.cnt AS val_b FROM
(SELECT id1, id2, COUNT(name) AS inter_cnt FROM
(SELECT c1.id AS id1, c2.id AS id2, c1.name AS name
FROM (SELECT c.nct_id AS id, c.name AS name FROM conditions AS c) AS c1
CROSS JOIN (select c.nct_id AS id, c.name AS name FROM conditions AS c) AS c2 ON c1.id <
c2.id AND c1.name = c2.name)
GROUP BY id1, id2)
AS ic
```

```
INNER JOIN
(SELECT c.id AS id, COUNT(c.name) AS cnt FROM (SELECT c.nct_id AS id, c.name AS name FROM
conditions AS c) AS c
GROUP BY c.id)
AS gc1 ON gc1.id = ic.id1
INNER JOIN
(SELECT c.id AS id, COUNT(c.name) AS cnt FROM (SELECT c.nct_id AS id, c.name AS name FROM
conditions AS c) AS c
GROUP BY c.id)
AS gc2 ON gc2.id = ic.id2)
AS j;

SELECT min(j.ja) from jindex as res;
```

0.0204

```
SELECT j.id1, j.id2 from jindex as j where j.jacard = 0.0204;
```

id1	id2
NCT02567396	NCT02595866
NCT02742597	NCT03002311

(b)
0.3653

```
SELECT ROUND( cnt_1/total,4) AS percentage FROM
(SELECT COUNT(ja) AS cnt_1 FROM jindex
WHERE ja = 1.0000)
,(SELECT COUNT(ja) AS total FROM jindex);
```

percentage
0.3653