**CS157A** **Homework 4**

**Points: 25**

**Question 1-5 (2 point)** **Question 6-10 (3 points)**

**Part 1**: SQL questions

The following queries are based on the campaign data (campaign-ca-2016.sql)

1. how many contributions are contained in the data?
   select count(contbr_nm) from campaign;

```
sqlite> select count(contbr_nm) from campaign;
count(contbr_nm)
----------------
180478
```

2. show the min and maximum amounts of all contributions?
   select min(contb_receipt_amt), max(contb_receipt_amt) from campaign;

```
min(contb_receipt_amt)  max(contb_receipt_amt)
----------------------  ----------------------
-10000                  10800
```

3. list the (distinct) ids and names of all the candidates in the data. order by name?
   select distinct(cand_id), cand_nm from campaign order by cand_nm asc;

```
sqlite> select distinct(cand_id), cand_nm from campaign order by cand_nm asc;
cand_id    cand_nm
---------  ----------------------
P60008059  Bush, Jeb
P60005915  Carson, Benjamin S.
P60008521  Christie, Christopher J.
P00003392  Clinton, Hillary Rodham
P60006111  Cruz, Rafael Edward 'Ted'
P60007242  Fiorina, Carly
P60007697  Graham, Lindsey O.
P80003478  Huckabee, Mike
P60008398  Jindal, Bobby
P60003670  Kasich, John R.
P60009685  Lessig, Lawrence
P60007671  O'Malley, Martin Joseph
P60007572  Pataki, George E.
P40003576  Paul, Rand
P20003281  Perry, James R. (Rick)
P60006723  Rubio, Marco
P60007168  Sanders, Bernard
P20002721  Santorum, Richard J.
P20003984  Stein, Jill
P80001571  Trump, Donald J.
P60006046  Walker, Scott
P60008885  Webb, James Henry Jr.
```

4. show the candidate name and number of contributions, for each candidate? order by number of contributions in descending order?
   select cand_nm, count(contbr_nm) from campaign group by cand_nm order by count(contbr_nm) desc;

```
sqlite> select cand_nm, count(contbr_nm) from campaign group by cand_nm order by count
(contbr_nm) desc;
cand_nm                   count(contbr_nm)
------------------------  ----------------
Sanders, Bernard          72179
Clinton, Hillary Rodham   42063
Cruz, Rafael Edward 'Ted' 21645
Carson, Benjamin S.       21045
Rubio, Marco              7994
Fiorina, Carly            4426
Paul, Rand                4117
Bush, Jeb                 2762
Kasich, John R.           701
Walker, Scott             670
Trump, Donald J.          590
Huckabee, Mike            447
O'Malley, Martin Joseph   383
Lessig, Lawrence          372
Graham, Lindsey O.        331
Christie, Christopher J.  316
Perry, James R. (Rick)    116
Webb, James Henry Jr.     106
Stein, Jill               85
Santorum, Richard J.      79
Jindal, Bobby             31
Pataki, George E.         20
```

5. show the candidate name and average contribution amount for each candidate, looking at positive contributions only? Order by average amount in descending order?
   select cand_nm, avg(contb_receipt_amt) from campaign where contb_receipt_amt > 0 group by cand_nm order by avg(contb_receipt_amt) desc;

```
sqlite> select cand_nm, avg(contb_receipt_amt) from campaign where contb_receipt_amt >
 0 group by cand_nm order by avg(contb_receipt_amt) desc;
cand_nm                   avg(contb_receipt_amt)
------------------------  ----------------------
Perry, James R. (Rick)    2060.77981651376
Christie, Christopher J.  1528.86688311688
Pataki, George E.         1522.5
Graham, Lindsey O.        1512.8125
Bush, Jeb                 1286.470866171
Walker, Scott             1046.54859504132
Kasich, John R.           1032.83410329986
O'Malley, Martin Joseph   765.523979057592
Jindal, Bobby             749.395483870968
Webb, James Henry Jr.     722.341132075472
Rubio, Marco              717.44661741214
Huckabee, Mike            547.378082191781
Clinton, Hillary Rodham   536.924861091049
Lessig, Lawrence          509.014824797844
Santorum, Richard J.      457.023797468354
Trump, Donald J.          365.379075342466
Fiorina, Carly            356.371891025641
Paul, Rand                206.038365056125
Cruz, Rafael Edward 'Ted' 160.517329784253
Stein, Jill               156.058823529412
Carson, Benjamin S.       141.50798943831
Sanders, Bernard          87.9972221912626
```

6. show the candidate name and the total amount received by each candidate. Order the output by total amount received.

select cand_nm, sum(contb_receipt_amt) from campaign group by cand_nm order by sum(contb_receipt_amt) desc;

```
sqlite> select cand_nm, sum(contb_receipt_amt) from campaign group by cand_nm order by
 sum(contb_receipt_amt) desc;
cand_nm                   sum(contb_receipt_amt)
------------------------  ----------------------
Clinton, Hillary Rodham   21899488.66
Sanders, Bernard          6109590.10000006
Rubio, Marco              4562345.79
Bush, Jeb                 3333245.23
Cruz, Rafael Edward 'Ted' 3133010.53
Carson, Benjamin S.       2741477.22
Fiorina, Carly            1452619.42
Paul, Rand                814093.720000002
Kasich, John R.           708585.37
Walker, Scott             461479.24
Christie, Christopher J.  449491
Graham, Lindsey O.        400495
O'Malley, Martin Joseph   290230.16
Huckabee, Mike            225351.6
Trump, Donald J.          208891.34
Perry, James R. (Rick)    208400
Lessig, Lawrence          186144.5
Webb, James Henry Jr.     76568.16
Santorum, Richard J.      36104.88
Pataki, George E.         30450
Jindal, Bobby             23231.26
Stein, Jill               13265
```

select cand_nm, sum(contb_receipt_amt) from campaign group by cand_nm order by sum(contb_receipt_amt) asc;

```
sqlite> select cand_nm, sum(contb_receipt_amt) from campaign group by cand_nm order by
 sum(contb_receipt_amt) asc;
cand_nm                   sum(contb_receipt_amt)
------------------------  ----------------------
Stein, Jill               13265
Jindal, Bobby             23231.26
Pataki, George E.         30450
Santorum, Richard J.      36104.88
Webb, James Henry Jr.     76568.16
Lessig, Lawrence          186144.5
Perry, James R. (Rick)    208400
Trump, Donald J.          208891.34
Huckabee, Mike            225351.6
O'Malley, Martin Joseph   290230.16
Graham, Lindsey O.        400495
Christie, Christopher J.  449491
Walker, Scott             461479.24
Kasich, John R.           708585.37
Paul, Rand                814093.720000002
Fiorina, Carly            1452619.42
Carson, Benjamin S.       2741477.22
Cruz, Rafael Edward 'Ted' 3133010.53
Bush, Jeb                 3333245.23
Rubio, Marco              4562345.79
Sanders, Bernard          6109590.10000006
Clinton, Hillary Rodham   21899488.66
```

**Part 2:**

Use the courses data (read the files `courses-ddl.sql` and `courses-small.sql`).

7. write an SQL query that gives the number of courses taken for every student in the student table.

   select name, count(distinct course_id) from student left outer join takes on student.ID = takes.ID group by name; (assume not including retaken courses)

```
sqlite> select name, count(distinct course_id) from student left outer join takes on s
tudent.ID = takes.ID group by name;
name      count(distinct course_id)
--------  -------------------------
Aoi       1
Bourikas  2
Brandt    1
Brown     2
Chavez    1
Levy      2
Peltier   1
Sanchez   1
Shankar   4
Snow      0
Tanaka    2
Williams  2
Zhang     2
```

   select name, count(course_id) from student left outer join takes on student.ID = takes.ID group by name; (assume including retaken courses)

```
sqlite> select name, count(course_id) from student left outer join takes on student.ID
 = takes.ID group by name;
name      count(course_id)
--------  ----------------
Aoi       1
Bourikas  2
Brandt    1
Brown     2
Chavez    1
Levy      3
Peltier   1
Sanchez   1
Shankar   4
Snow      0
Tanaka    2
Williams  2
Zhang     2
```

8. For each instructor, show the instructor name and the number of sections that have been taught by that instructor. You do not need to include instructors who have never taught a section. List in order of decreasing number of sections taught.

   select name, count(sec_id) from instructor natural join teaches group by name order by count(sec_id) desc;

```
sqlite> select name, count(sec_id) from instructor natural join teaches group by name
order by count(sec_id) desc;
name        count(sec_id)
----------  -------------
Srinivasan  3
Brandt      3
Katz        2
Crick       2
Wu          1
Mozart      1
Kim         1
El Said     1
Einstein    1
```

9. Give the number of semester/year combinations in which sections have been offered.

   select count(*) as '# of sem/year combos' from (select distinct semester, year from section order by year);

```
sqlite> select count(*) as '# of sem/year combos' from (select distinct semester, year
 from section order by year);
# of sem/year combos
--------------------
5
```

**Part 3:**

The following queries are based on the campaign data (campaign-ca-2016.sql)

```
create table candidate (
  cand_id      varchar(12) primary key,      -- cand_id
  name         varchar(40)                   -- cand_nm
);
```

10. Write an insert statement to fill the candidate table from Campaign table. The candidate table, after it is filled, should contain one row for each candidate (there are 22 candidates in the campaign table). -- Hint: use the kind of insert statement in which you list the fields to be filled in parentheses after the table name

insert into candidate (cand_id, name) select distinct cand_id, cand_nm from campaign;

```
sqlite> insert into candidate (cand_id, name) select distinct cand_id, cand_nm from ca
mpaign;
sqlite> select * from candidate;
cand_id    name
---------  -------------------------
P60007168  Sanders, Bernard
P60006723  Rubio, Marco
P60008521  Christie, Christopher J.
P60008059  Bush, Jeb
P60005915  Carson, Benjamin S.
P00003392  Clinton, Hillary Rodham
P60003670  Kasich, John R.
P80001571  Trump, Donald J.
P60007242  Fiorina, Carly
P60006111  Cruz, Rafael Edward 'Ted'
P40003576  Paul, Rand
P20002721  Santorum, Richard J.
P60008398  Jindal, Bobby
P20003984  Stein, Jill
P80003478  Huckabee, Mike
P60009685  Lessig, Lawrence
P60008885  Webb, James Henry Jr.
P60007572  Pataki, George E.
P60007671  O'Malley, Martin Joseph
P60007697  Graham, Lindsey O.
P20003281  Perry, James R. (Rick)
P60006046  Walker, Scott
```