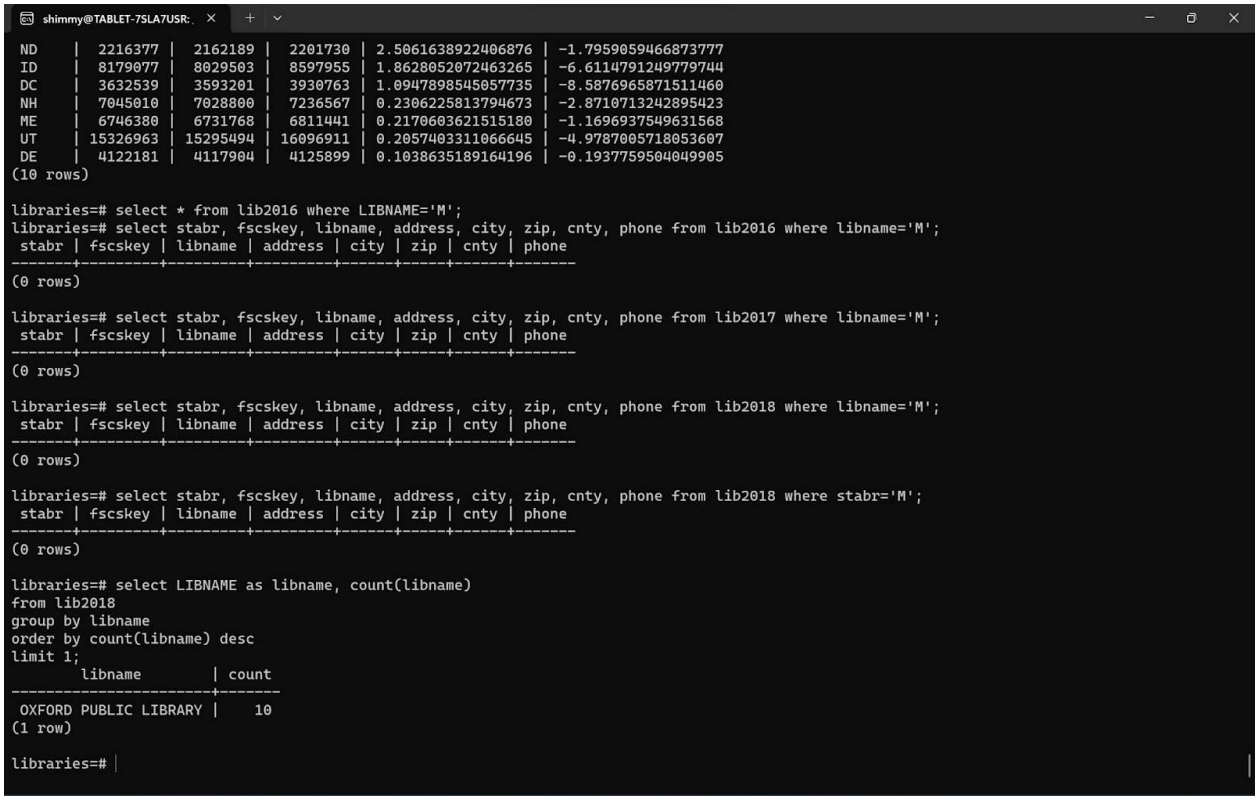
Public Libraries Assignment

Query 1: “What is the most common library name in the 2018 data-set?”

Your answer should be a tuple with two columns, in this order: libname, count.

```
select LIBNAME as libname, count(libname)
from lib2018
group by libname
order by count(libname) desc
limit 1;
```

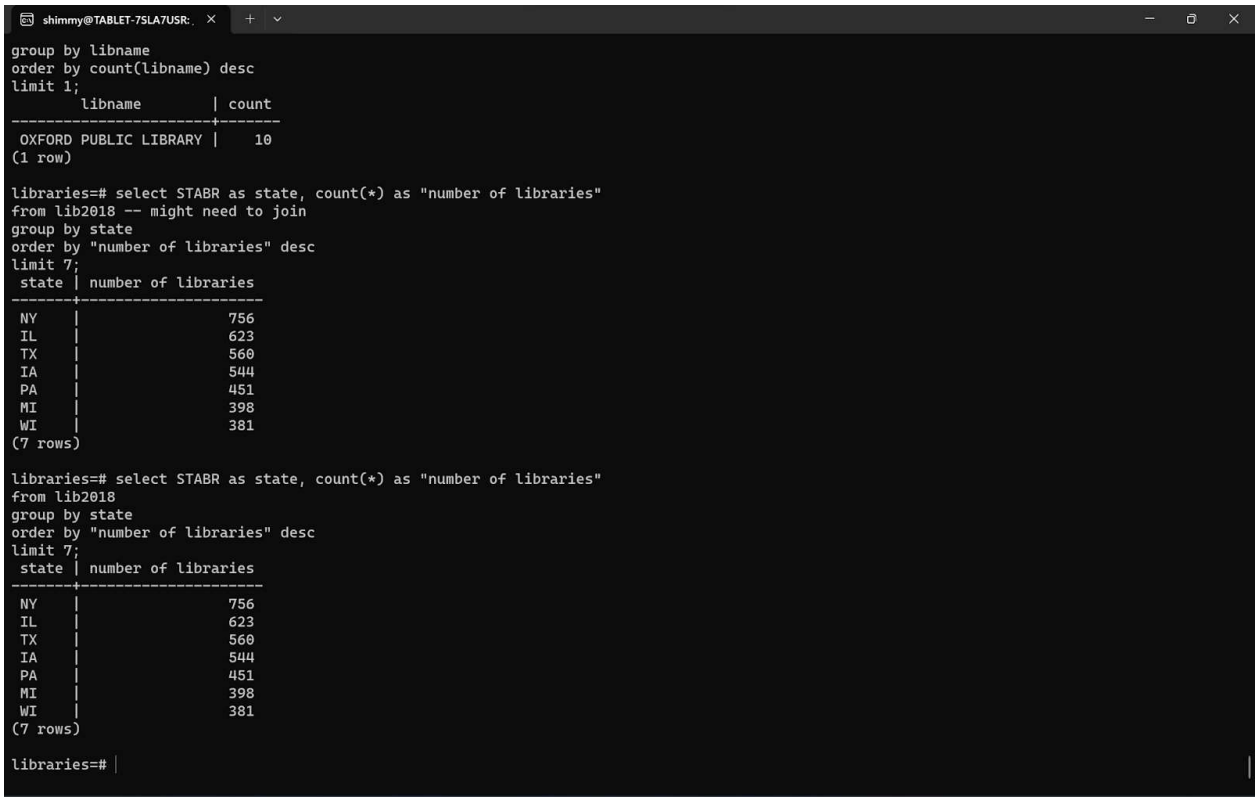


Query 2: “Which state has the most libraries?”

Your result set should contain seven tuples, in descending order of “number of libraries”. Each tuple should contain (and be labeled), in this order: state, number of libraries

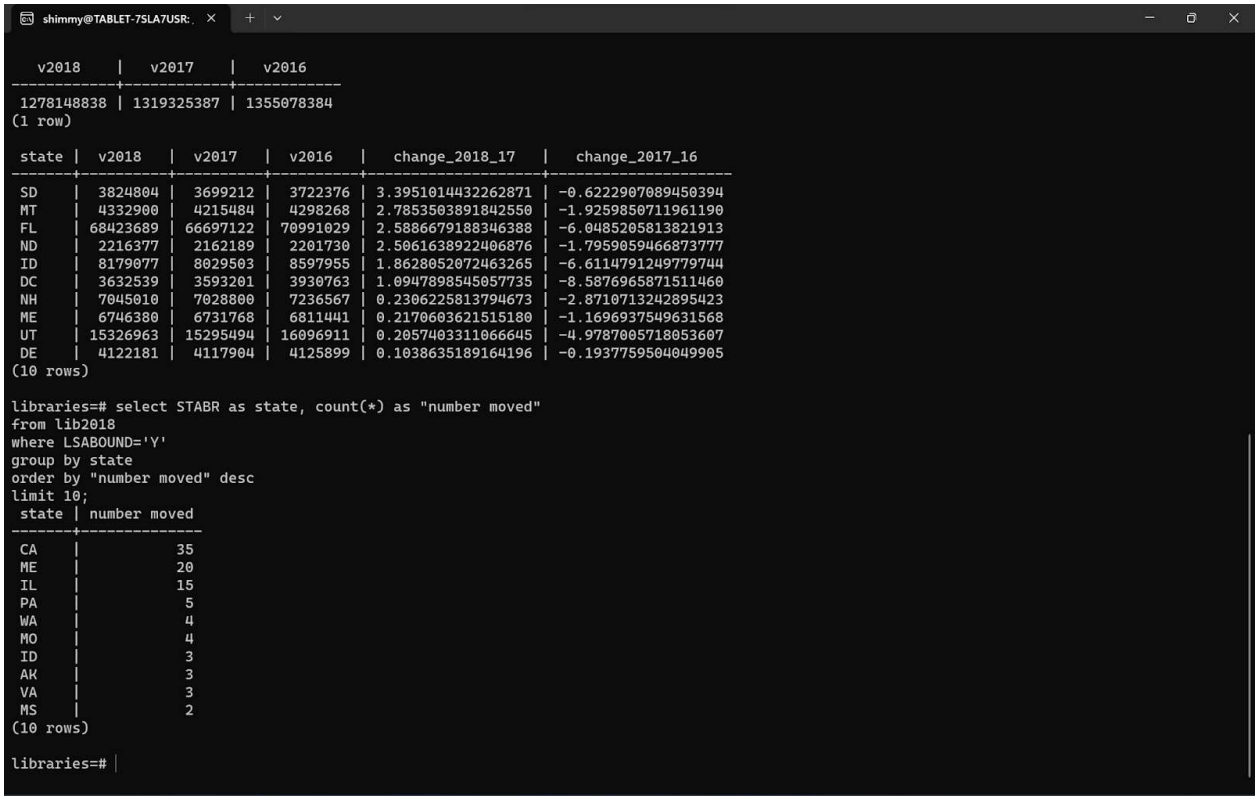
State is represented by the 2-letter code. Use the 2018 dataset.

```
select STABR as state, count(*) as "number of libraries"
from lib2018
group by state
order by "number of libraries" desc
limit 7;
```



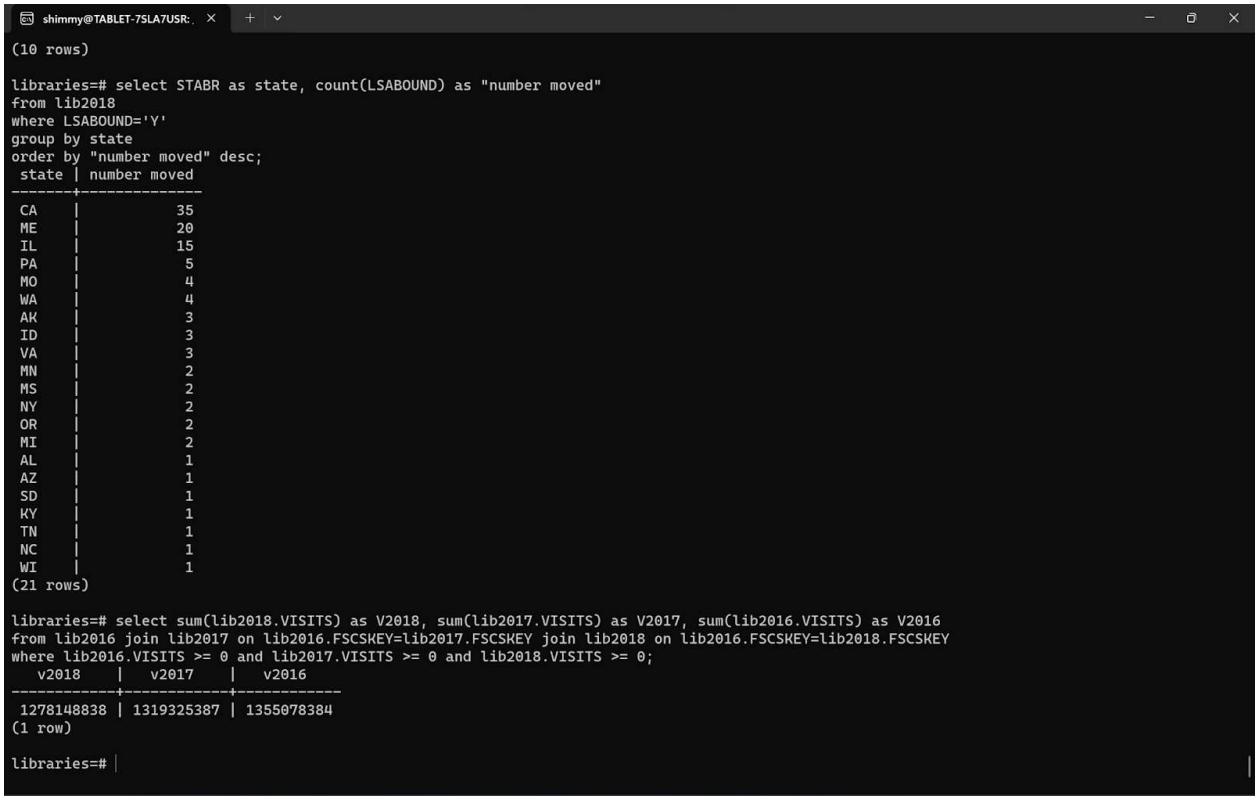
Query 3: “For each state, in 2018, how many libraries changed their address in any way?” Your result set should contain, in descending order of “number libraries moved in that state in 2018”. Each tuple should contain (and be labeled as) in this order: state, number moved.

```
select STABR as state, count(*) as "number moved"
from lib2018
where LSABOUND='Y'
group by state
order by "number moved" desc
limit 10;
```



Query 4: “Return the number of visits to libraries in 2018, 2017, and 2016.”
The query is intended to get a sense of whether library usage has increased, decreased, or stayed roughly the same over this time period. Therefore: you need to do an “apples-to-apples” comparison, such that only libraries that were open in each of these years are used in the computation. Be sure to only include “valid survey responses/data” (see above) in the computation. Note: you’re aggregating data across the United States as a whole.
The result set should be a single tuple, containing (and labelled), in this order: V2018, V2017, V2016.

```
select sum(lib2018.VISITS) as V2018, sum(lib2017.VISITS) as
V2017, sum(lib2016.VISITS) as V2016
from lib2016 join lib2017 on lib2016.FSCSKEY=lib2017.FSCSKEY
join lib2018 on lib2016.FSCSKEY=lib2018.FSCSKEY
where lib2016.VISITS >= 0 and lib2017.VISITS >= 0 and
lib2018.VISITS >= 0;
```



5. “Do a by-state analysis of the above query (details below).”

Instead of reporting data aggregated across the United States (the previous query), you’re aggregating data by state. You’ll report the “raw number of visits” for each of the three years. In addition: you’ll report (by state), the trends of visits: specifically, the percentage change (whether positive or negative) in 2018 relative to 2017, and the percentage change in 2017 relative to 2016.

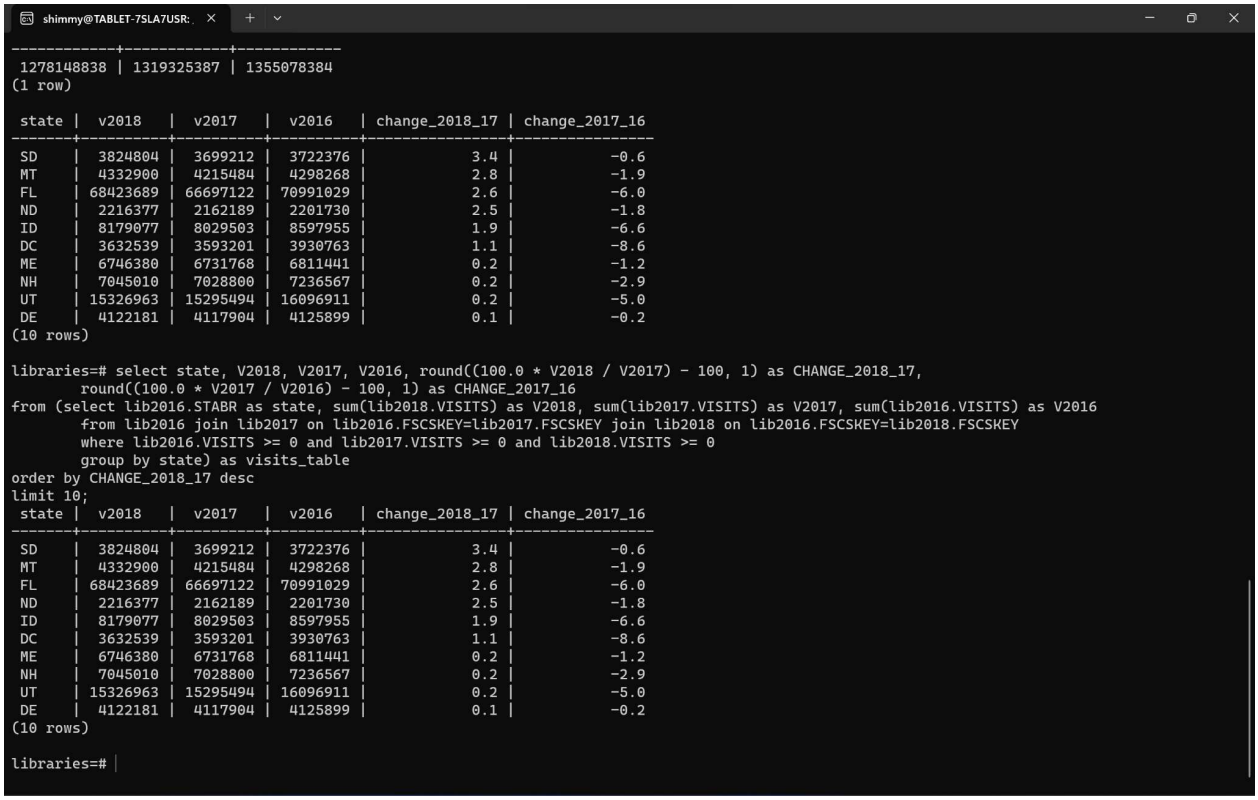
Your answer should be a tuple containing (and labeled) in this order: (a) state (b) V2018 (c) V2017 (d) V2016 (e) CHANGE_2018_17 (f) CHANGE_2017_16

Report only the first ten tuples when the result set is ordered by descending “percentage change from 2018 to 2017” values.

Note: this query uses the previous queries’ semantics with respect to e.g., library usage.

Suggestion: be careful about integer division!

```
select state, V2018, V2017, V2016, round((100.0 * V2018 / V2017)
- 100, 1) as CHANGE_2018_17, round((100.0 * V2017 / V2016) -
100, 1) as CHANGE_2017_16
from (select lib2016.STABR as state, sum(lib2018.VISITS) as
V2018, sum(lib2017.VISITS) as V2017, sum(lib2016.VISITS) as
V2016
     from lib2016 join lib2017 on
lib2016.FSCSKEY=lib2017.FSCSKEY join lib2018 on
lib2016.FSCSKEY=lib2018.FSCSKEY
     where lib2016.VISITS >= 0 and lib2017.VISITS >= 0 and
lib2018.VISITS >= 0
     group by state) as visits_table
order by CHANGE_2018_17 desc
limit 10;
```


shimmy@TABLET-7SLA7USR:
------------+------------+------------
1278148838 | 1319325387 | 1355078384
(1 row)
state | v2018 | v2017 | v2016 | change_2018_17 | change_2017_16
SD | 3824804 | 3699212 | 3722376 | 3.4 | -0.6
MT | 4332900 | 4215484 | 4298268 | 2.8 | -1.9
FL | 68423689 | 66697122 | 70991029 | 2.6 | -6.0
ND | 2216377 | 2162189 | 2201730 | 2.5 | -1.8
ID | 8179077 | 8029503 | 8597955 | 1.9 | -6.6
DC | 3632539 | 3593201 | 3930763 | 1.1 | -8.6
ME | 6746380 | 6731768 | 6811441 | 0.2 | -1.2
NH | 7045010 | 7028800 | 7236567 | 0.2 | -2.9
UT | 15326963 | 15295494 | 16096911 | 0.2 | -5.0
DE | 4122181 | 4117904 | 4125899 | 0.1 | -0.2
(10 rows)
libraries=# select state, V2018, V2017, V2016, round((100.0 * V2018 / V2017) - 100, 1) as CHANGE_2018_17,
round((100.0 * V2017 / V2016) - 100, 1) as CHANGE_2017_16
from (select lib2016.STABR as state, sum(lib2018.VISITS) as V2018, sum(lib2017.VISITS) as V2017, sum(lib2016.VISITS) as V2016
from lib2016 join lib2017 on lib2016.FSCSKEY=lib2017.FSCSKEY join lib2018 on lib2016.FSCSKEY=lib2018.FSCSKEY
where lib2016.VISITS >= 0 and lib2017.VISITS >= 0 and lib2018.VISITS >= 0
group by state) as visits_table
order by CHANGE_2018_17 desc
limit 10;
state | v2018 | v2017 | v2016 | change_2018_17 | change_2017_16
SD | 3824804 | 3699212 | 3722376 | 3.4 | -0.6
MT | 4332900 | 4215484 | 4298268 | 2.8 | -1.9
FL | 68423689 | 66697122 | 70991029 | 2.6 | -6.0
ND | 2216377 | 2162189 | 2201730 | 2.5 | -1.8
ID | 8179077 | 8029503 | 8597955 | 1.9 | -6.6
DC | 3632539 | 3593201 | 3930763 | 1.1 | -8.6
ME | 6746380 | 6731768 | 6811441 | 0.2 | -1.2
NH | 7045010 | 7028800 | 7236567 | 0.2 | -2.9
UT | 15326963 | 15295494 | 16096911 | 0.2 | -5.0
DE | 4122181 | 4117904 | 4125899 | 0.1 | -0.2
(10 rows)
libraries=#