

Last Name:

Suh

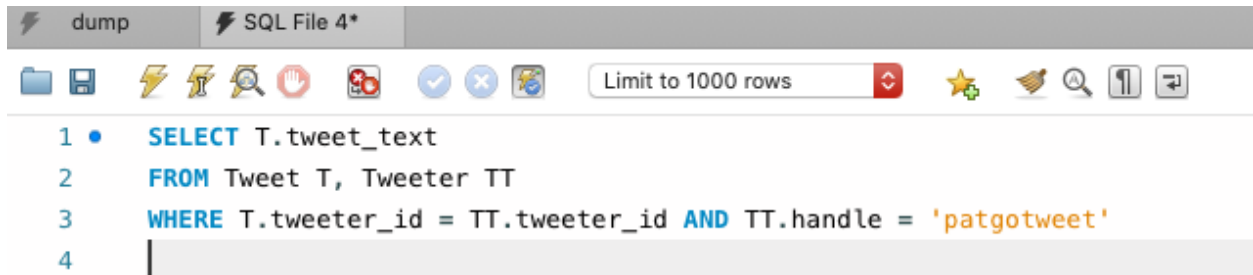
First Name:

Joowon

Student ID: 44414081

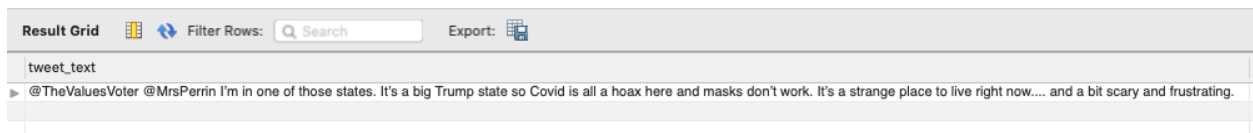
1. [10 pts] Find the text of all tweets that were posted by the tweeter with the handle 'patgotweet'.

a) [7 pts] SQL Query:



```
1 • SELECT T.tweet_text
2 FROM Tweet T, Tweeter TT
3 WHERE T.tweeter_id = TT.tweeter_id AND TT.handle = 'patgotweet'
4 |
```

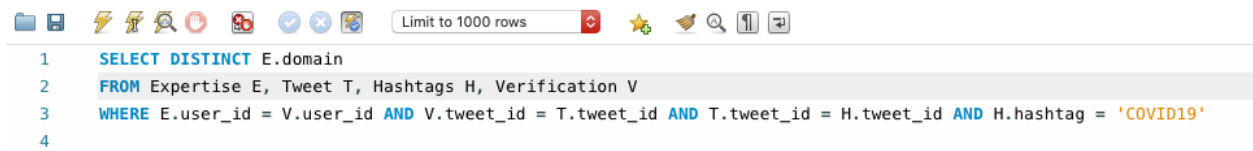
b) [3 pts] Result: (1 Row)



tweet_text
► @TheValuesVoter @MrsPerrin I'm in one of those states. It's a big Trump state so Covid is all a hoax here and masks don't work. It's a strange place to live right now.... and a bit scary and frustrating.

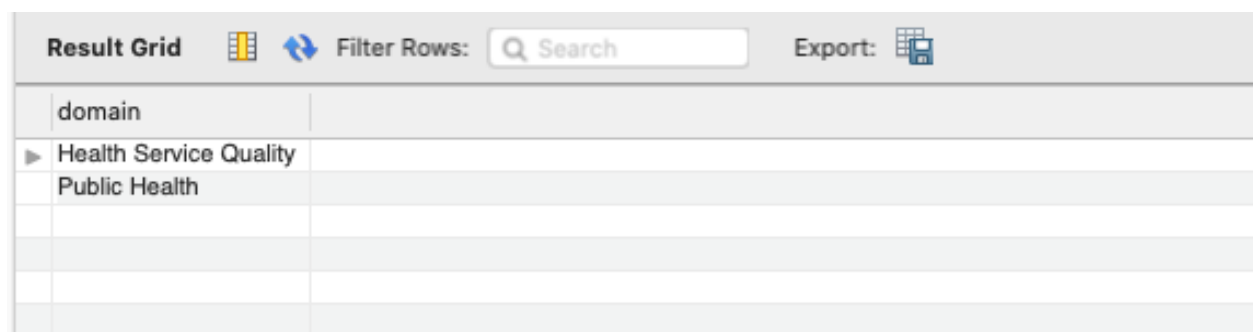
2. [10 pts] List the **distinct** domains of expertise for checkers who have verified tweets that have the hashtag “COVID19”. (Note: The hashtag value is all in capital letters.)


a) [7 pts] SQL Query:



```
1 SELECT DISTINCT E.domain
2 FROM Expertise E, Tweet T, Hashtags H, Verification V
3 WHERE E.user_id = V.user_id AND V.tweet_id = T.tweet_id AND T.tweet_id = H.tweet_id AND H.hashtag = 'COVID19'
4
```

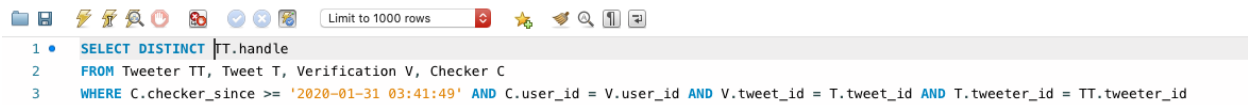
b) [3 pts] Result: (2 Rows)



Result Grid		Filter Rows: <input type="text" value="Search"/>	Export: 
	domain		
▶	Health Service Quality		
	Public Health		

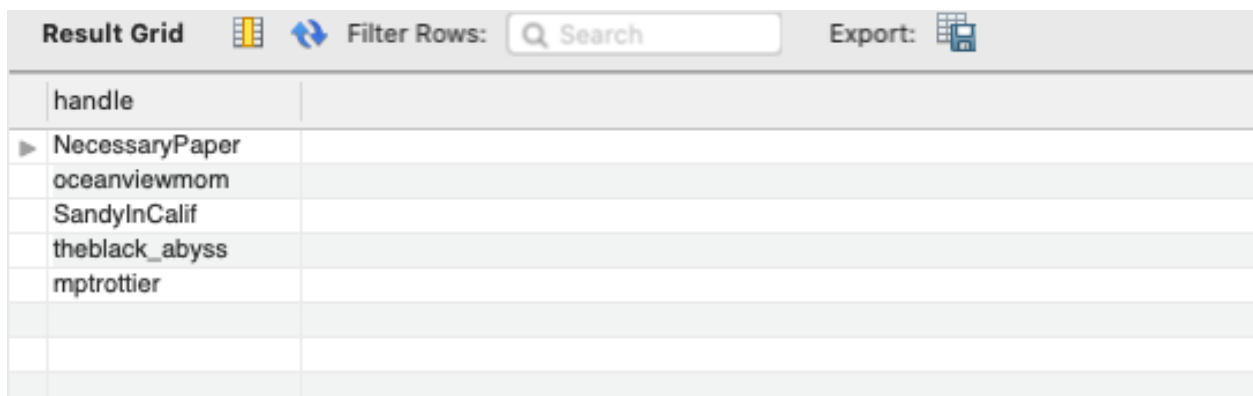
3. [10 pts] List the handles of Tweeters who have posted a tweet that has been verified by a Checker who started as a checker after the date “2020-01-31 03:41:49”.

a) [7 pts] SQL Query:



```
1 • SELECT DISTINCT TT.handle
2 FROM Tweeter TT, Tweet T, Verification V, Checker C
3 WHERE C.checker_since >= '2020-01-31 03:41:49' AND C.user_id = V.user_id AND V.tweet_id = T.tweet_id AND T.tweeter_id = TT.tweeter_id
```

b) [3 pts] Result: (5 Rows)



handle	
NecessaryPaper	
oceanviewmom	
SandyInCalif	
theblack_abyss	
mptrottier	

4. [15 pts] For verified tweets that contain the hashtag "COVID19", find the associated evidence URLs, verification comments, and checkers' first and last names (**Again: "COVID19" is in all caps.**)

a) [12 pts] SQL Query:

```

1 SELECT E.url, V.comment, U.name_first, U.name_last
2 FROM User U, Evidence E, Hashtags H, Checker C, Verification V, Tweet T, VerifiedUsing VU
3 WHERE T.tweet_id = V.tweet_id AND V.user_id = C.user_id AND C.user_id = U.user_id AND H.tweet_id = T.tweet_id AND V.ver_id = VU.ver_id AND VU.ev_id = E.ev_id AND H.hashtag = 'COVID19'

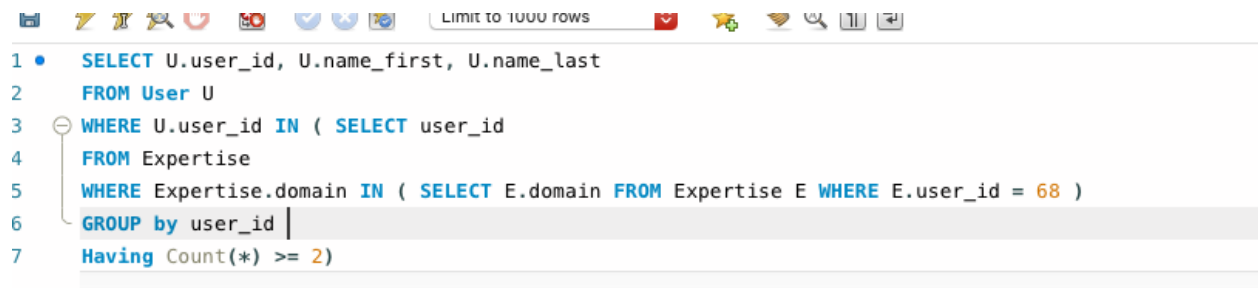
```

b) [3 pts] Result (12 rows):

Result Grid		Filter Rows:	Search	Export:
url	comment	name_first	name_last	
▶ http://states-covid-numbers.org	Masks works! Check the CDC	Jonathan	Howard	
http://florida-covid19.gov	Masks works! Check the CDC	Jonathan	Howard	
https://cdc.gov	Masks works! Check the CDC	Jonathan	Howard	
http://florida-covid19.gov	Masks works! Check the CDC	Antonio	Olson	
https://cdc.gov	Masks works! Check the CDC	Antonio	Olson	
http://states-covid-numbers.org	Masks works! Check the CDC	Antonio	Olson	
http://states-covid-numbers.org	Masks works! Check the CDC	Gina	Miranda	
http://florida-covid19.gov	Masks works! Check the CDC	Gina	Miranda	
https://cdc.gov	Masks works! Check the CDC	Gina	Miranda	
https://cdc.gov	Masks works! Check the CDC	Courtney	White	
http://mask-works.info	Masks works! Check the CDC	Courtney	White	
http://covid-is-not-hoax.net	Masks works! Check the CDC	Courtney	White	

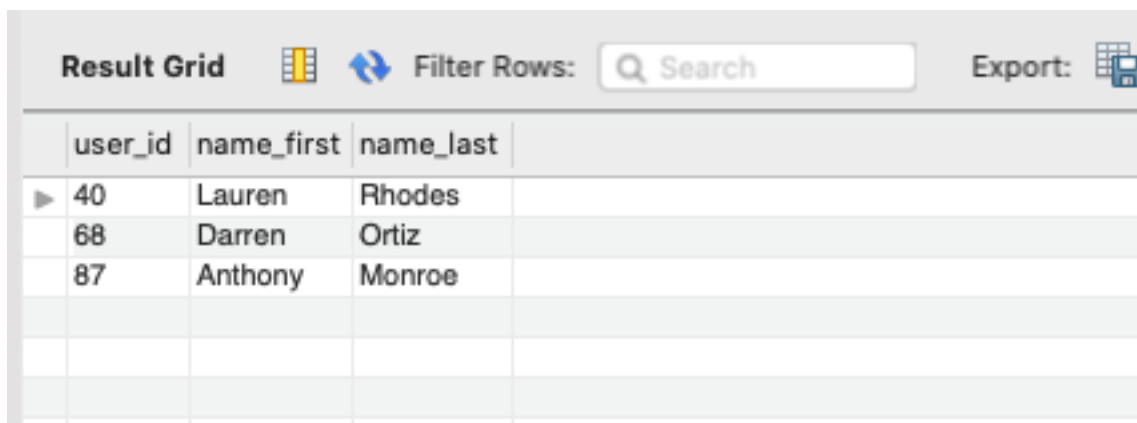
5. [15 pts] Find the user IDs, first names, and last names of checkers that have **all** the domains of expertise from the user with ID = 68. (Note: Your answer will include the "ID = 68" checker as well, of course.)

a) [12 pts] SQL Query:



```
1 • SELECT U.user_id, U.name_first, U.name_last
2   FROM User U
3  WHERE U.user_id IN ( SELECT user_id
4                       FROM Expertise
5                       WHERE Expertise.domain IN ( SELECT E.domain FROM Expertise E WHERE E.user_id = 68 )
6                       GROUP by user_id |
7                       Having Count(*) >= 2)
```

b) [3 pts] Result: (3 Rows)



	user_id	name_first	name_last
▶	40	Lauren	Rhodes
	68	Darren	Ortiz
	87	Anthony	Monroe


a) [7 pts] SQL Query:

b) [3 pts] Result: (2 Rows)

[illegible]

7. [15 pts] Find tweet ids and the number of replies for each tweet that has one or more replies. List only the top five tweets that have the highest number of replies.

a) [12 pts] SQL Query:



```
1 • select t.replied_to_tweet, count(*)
2   from Tweet t
3  group by t.replied_to_tweet
4  having 0 < ( select count(*) from Tweet t2 where t.replied_to_tweet = t2.replied_to_tweet )
5  order by count(*) DESC
6  limit 5
```

b) [3 pts] Result: (5 Rows)

	replied_to_tweet	count(*)
▶	1321293727105765376	3
	1321470312509300738	3
	1321418974886854656	2
	1321457297441214464	2
	1321493086120210432	2

8. [15 pts] For tweets that have two or more reactions (replies and/or quotes), print their tweet id along with their number of replies and number of quotes. (Note that for such tweets, the sum of replies and quotes should be 2 or more). Order the result by the number of reactions in largest-first order.

a) [12 pts] SQL Query:

```

1 • select rtable.replied_to_tweet as tweet_id, qtable.numq + rtable.numrep as count
2 from ( select t.quoted_tweet, count(*) as numq from Tweet t group by t.quoted_tweet order by count(*) DESC limit 5 ) as qtable,
3 (select t.replied_to_tweet, count(*) as numrep from Tweet t group by t.replied_to_tweet having 0 < ( select count(*) from Tweet t2 where t.replied_to_tweet = t2.replied_to_tweet ) order by count(*) DESC ) as rtable
4 where rtable.replied_to_tweet = qtable.quoted_tweet
5 union
6 select rtable.replied_to_tweet as tweet_id, rtable.numrep
7 from ( select t.quoted_tweet, count(*) as numq from Tweet t group by t.quoted_tweet order by count(*) DESC limit 5 ) as qtable,
8 (select t.replied_to_tweet, count(*) as numrep from Tweet t group by t.replied_to_tweet having 0 < ( select count(*) from Tweet t2 where t.replied_to_tweet = t2.replied_to_tweet ) order by count(*) DESC ) as rtable
9 where rtable.numrep > 1
10 order by count desc

```

b) [3 pts] Result (9 rows):

	tweet_id	count	
►	1321293727105765376	3	
	1321470312509300738	3	
	1321211561046933514	2	
	1321418974886854656	2	
	1321457297441214464	2	
	1321493086120210432	2	
	1321494210185342976	2	
	1321496681217548288	2	
	1321497818146635776	2	



