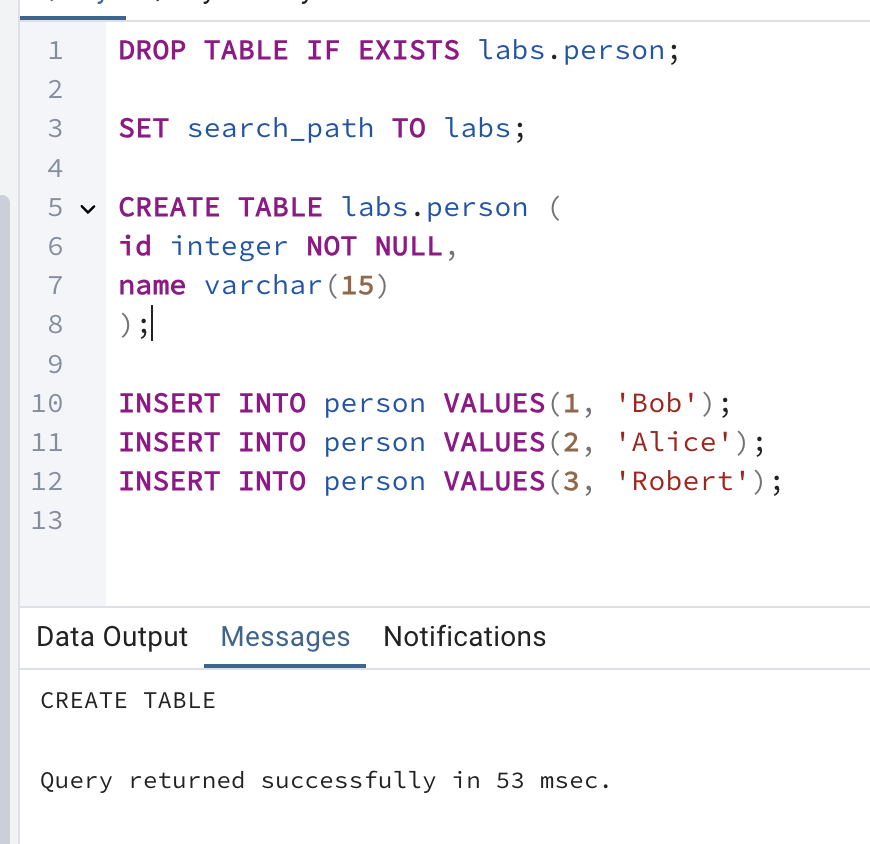
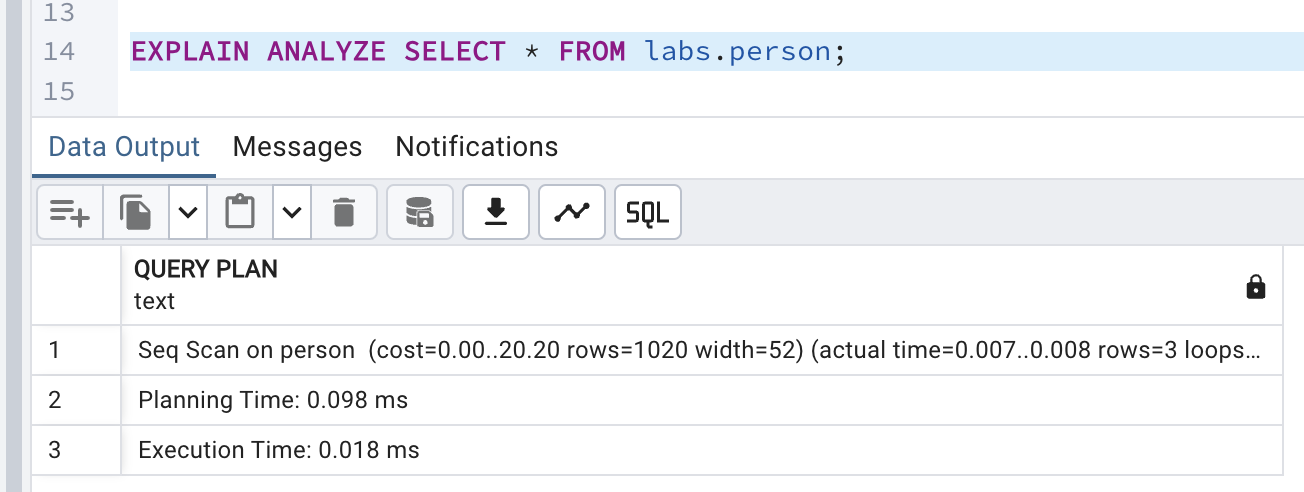
1. Created table



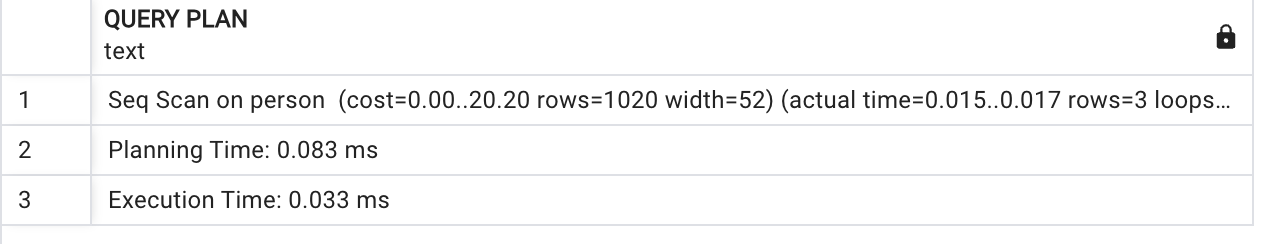
It shows a detailed execution plan of a query, helping to understand how the database process it, things like scans, joins.

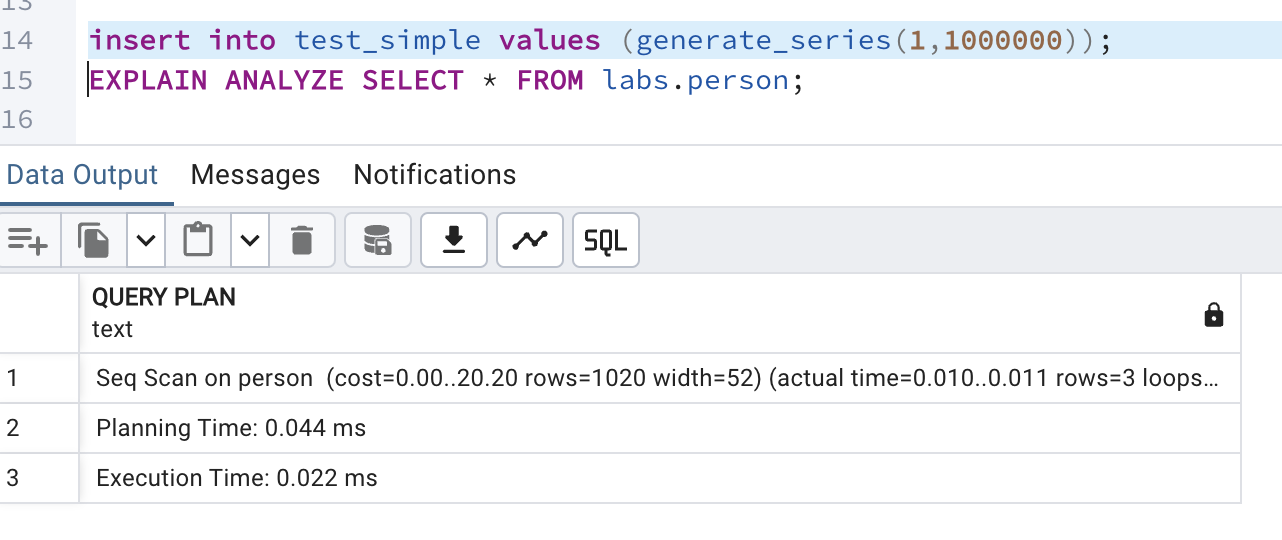


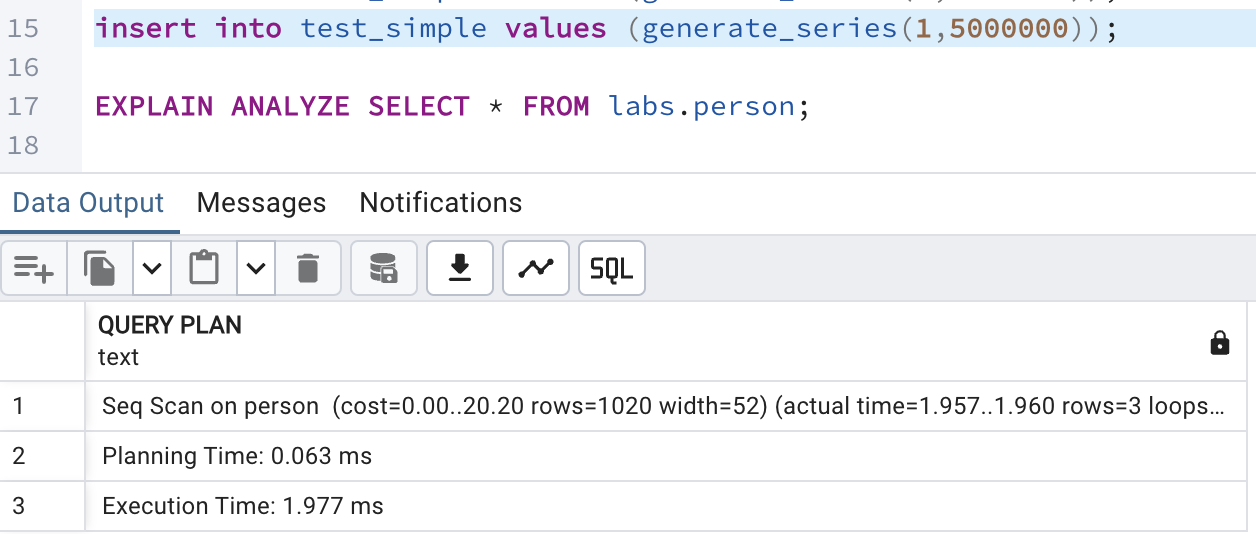
2.

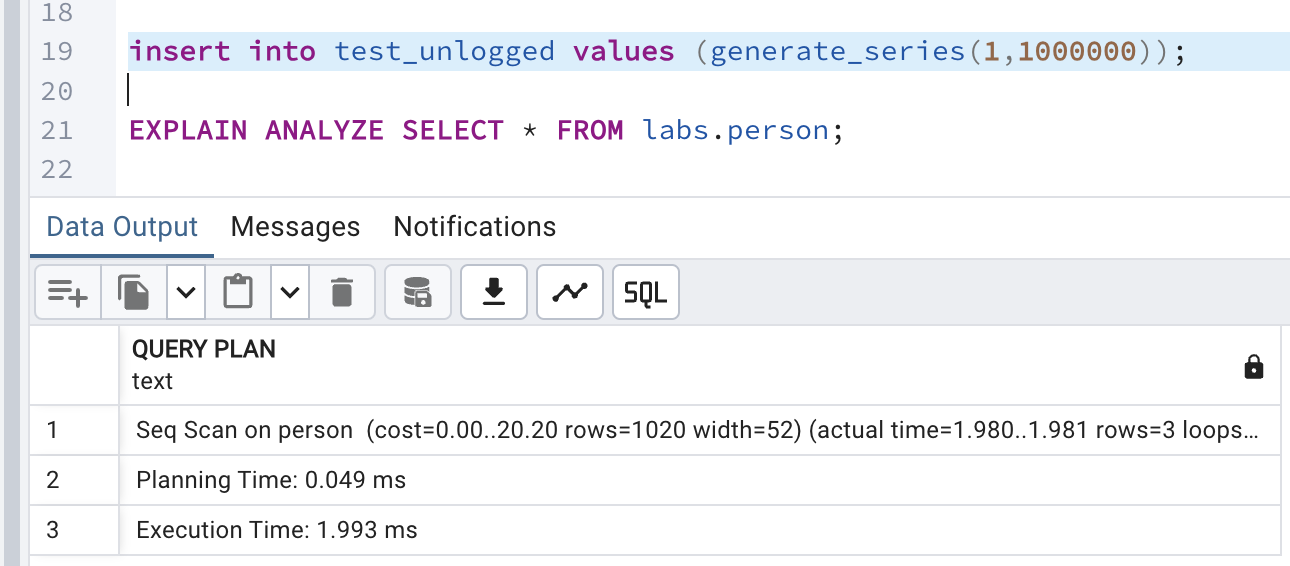
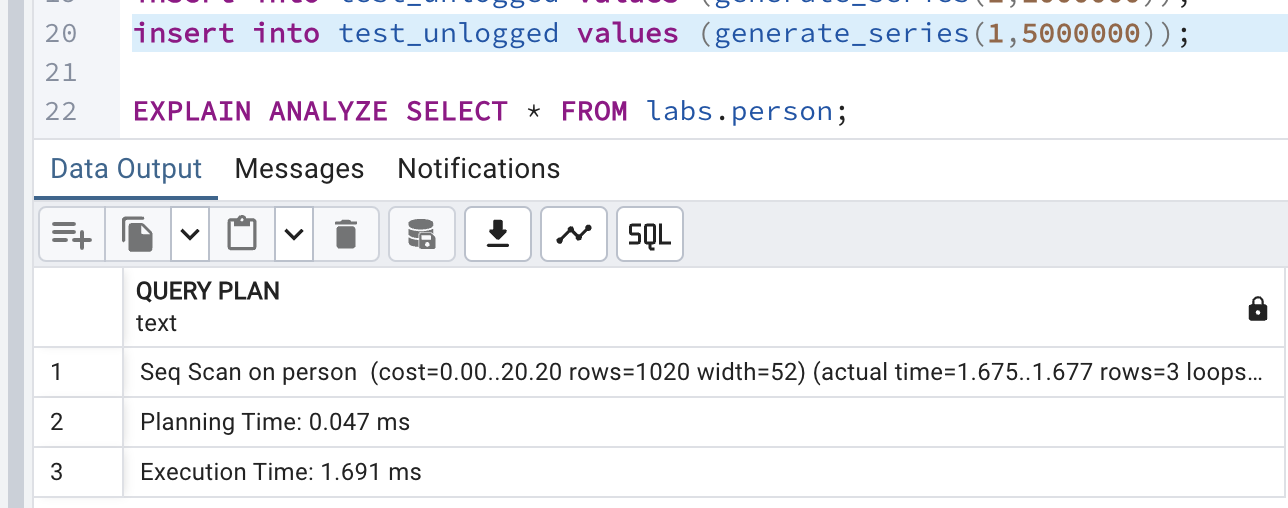
Time on inserting first.

labs.test\_simple(a int,b int)



The first query was very fast, but the second one was much slower. This might be because the database had to do more work after adding a lot of new data.





test\_simple was much slower after inserting 1,500,000 rows, taking 1.977 ms

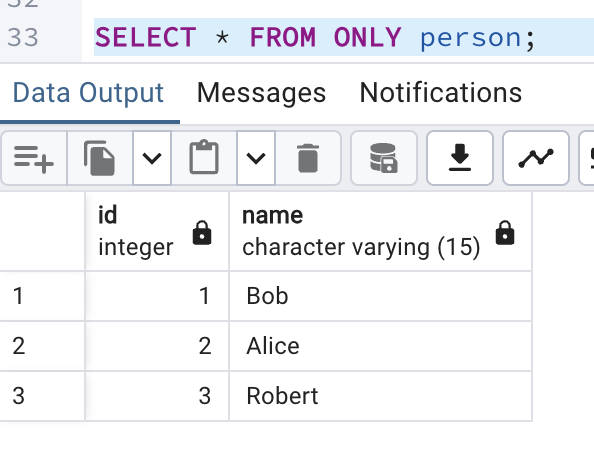
test\_unlogged performed better, with 1,500,000 rows taking 1.691 ms

For test\_unlogged, inserting 1,000,000 rows later took 1.993 ms, still faster than test\_simple.

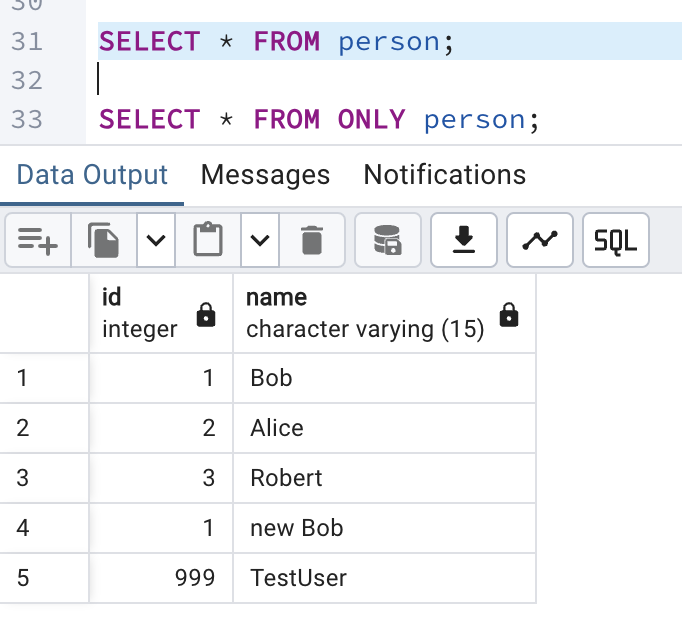
Unlogged tables are faster for inserts because they skip WAL logging, but they risk data loss.

2.2

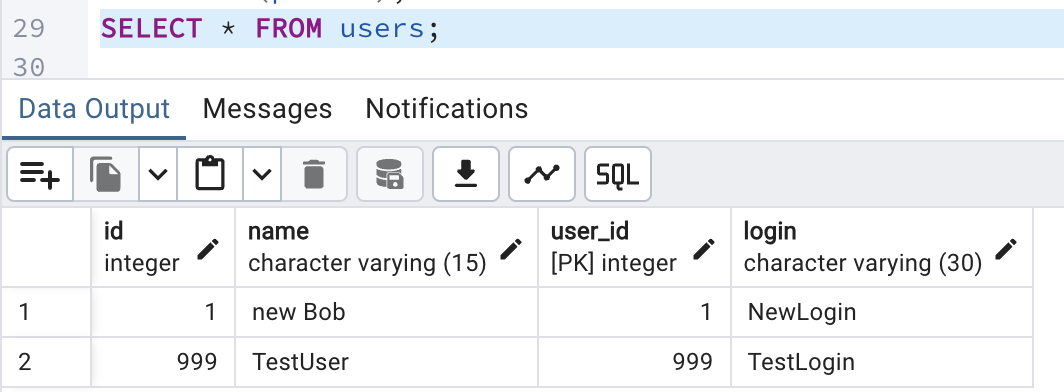
Select lab person only shows originally inserted member, with only keyword with it

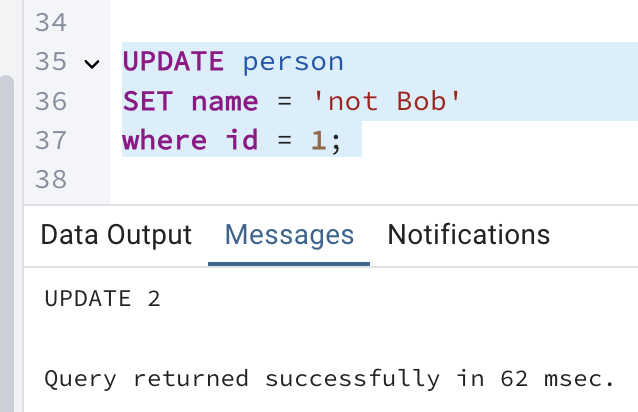


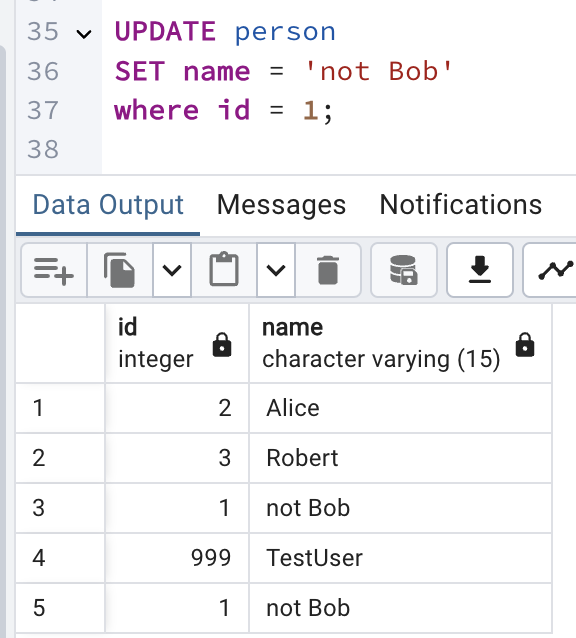
Without only keyword, it only shows the all members ids, the ones who also where inserted in users, because user inherits persons table.

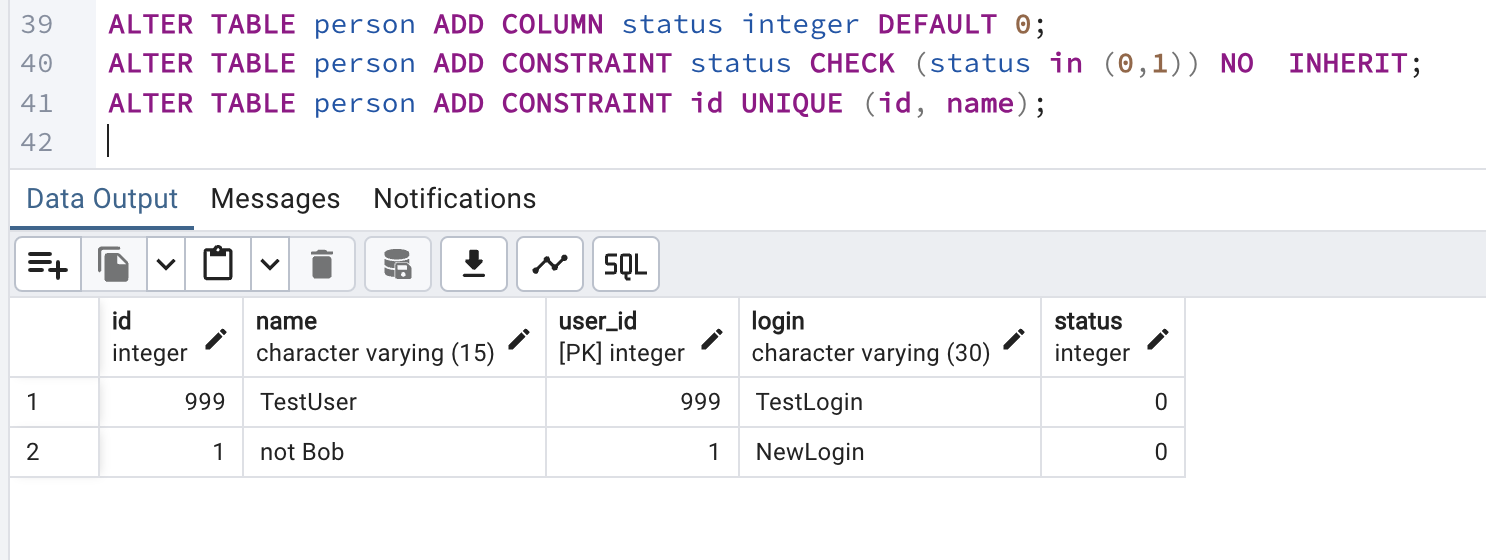


Selecting from users shows inserted members into users table, nothing more nothing less, just the table and its rows.



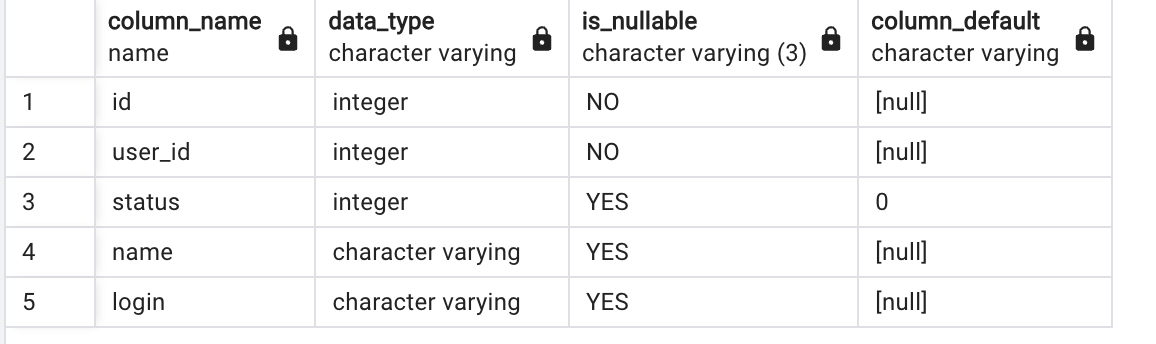


In order to change update only for the person table only we update it like this: 



I can see that by altering the person table, the users table was altered as well

Constraints for person table, like id and name should be unique and status should be checked

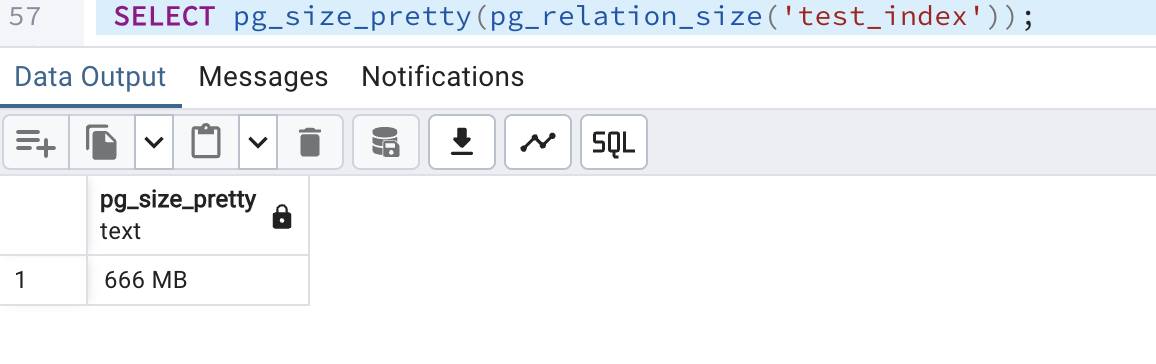


Users added status after altering operations, it added attributes from person and constraint has not been added to the user though. Seems like constraint can be only done to the table and does not affect child

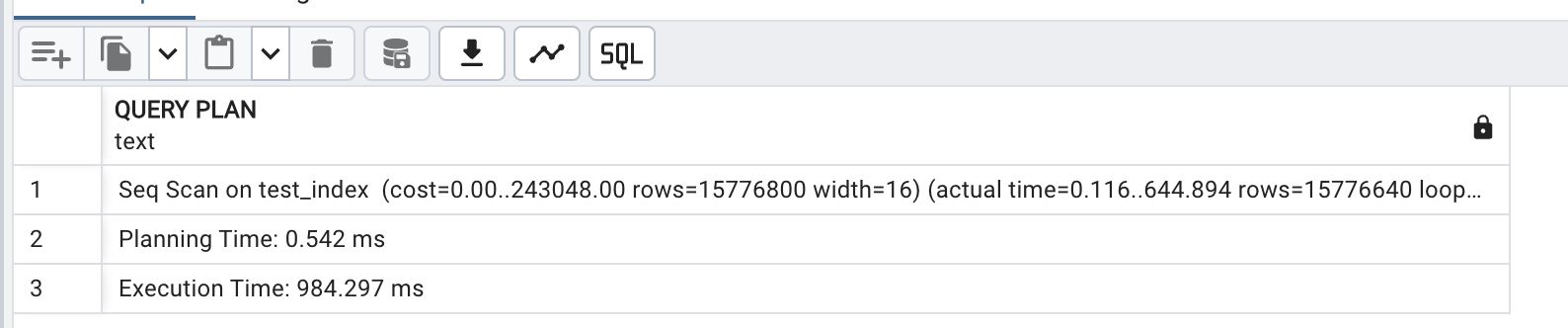
3.1

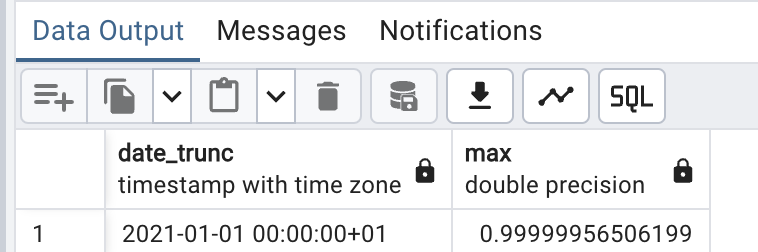


Checking the table size:

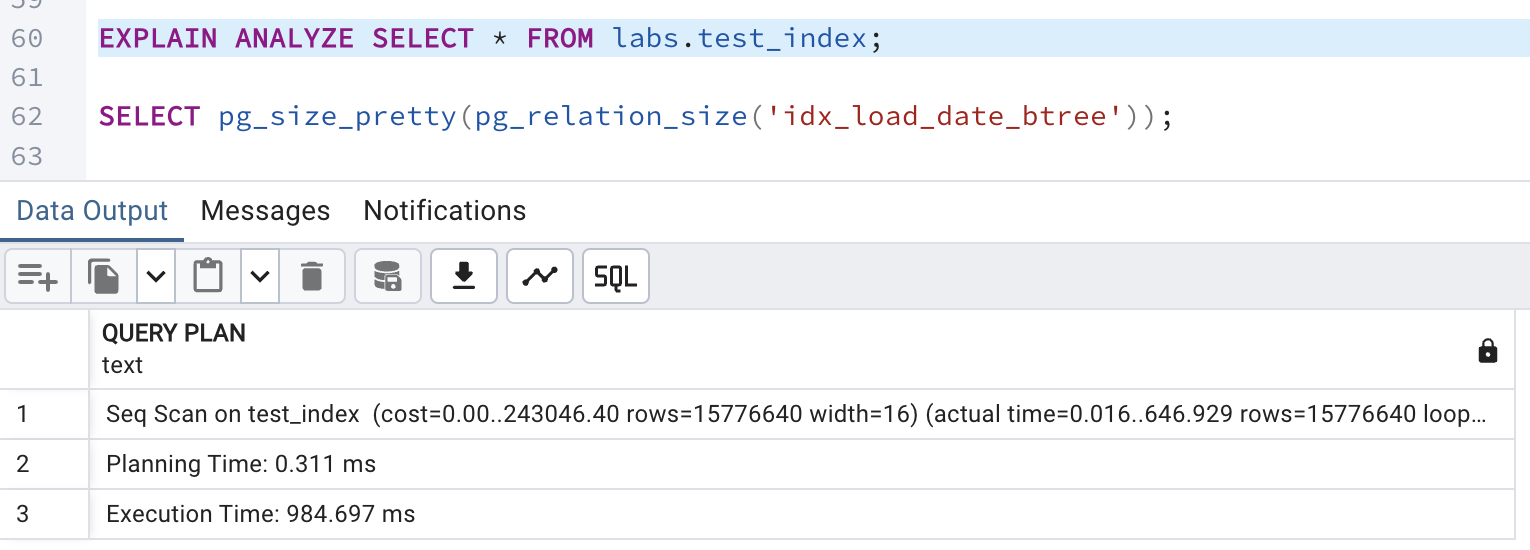


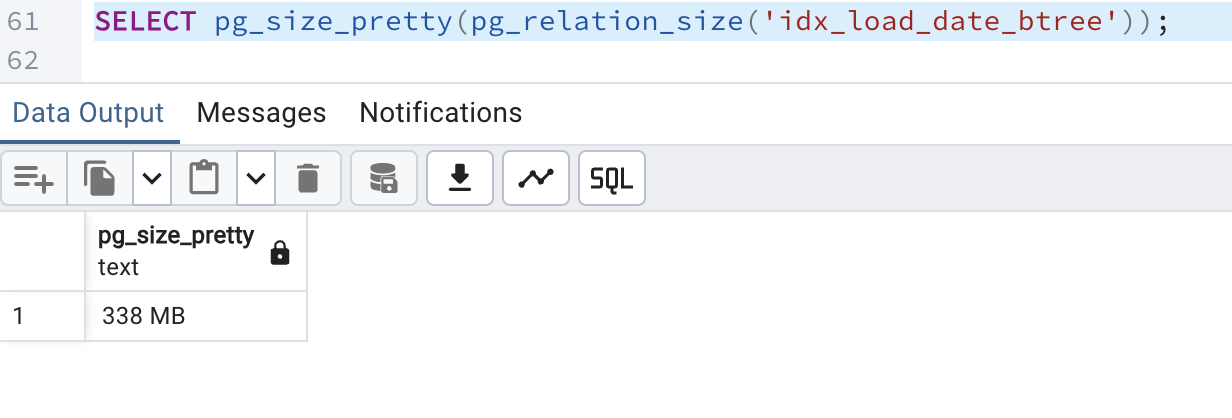
it is how much time it took to run select



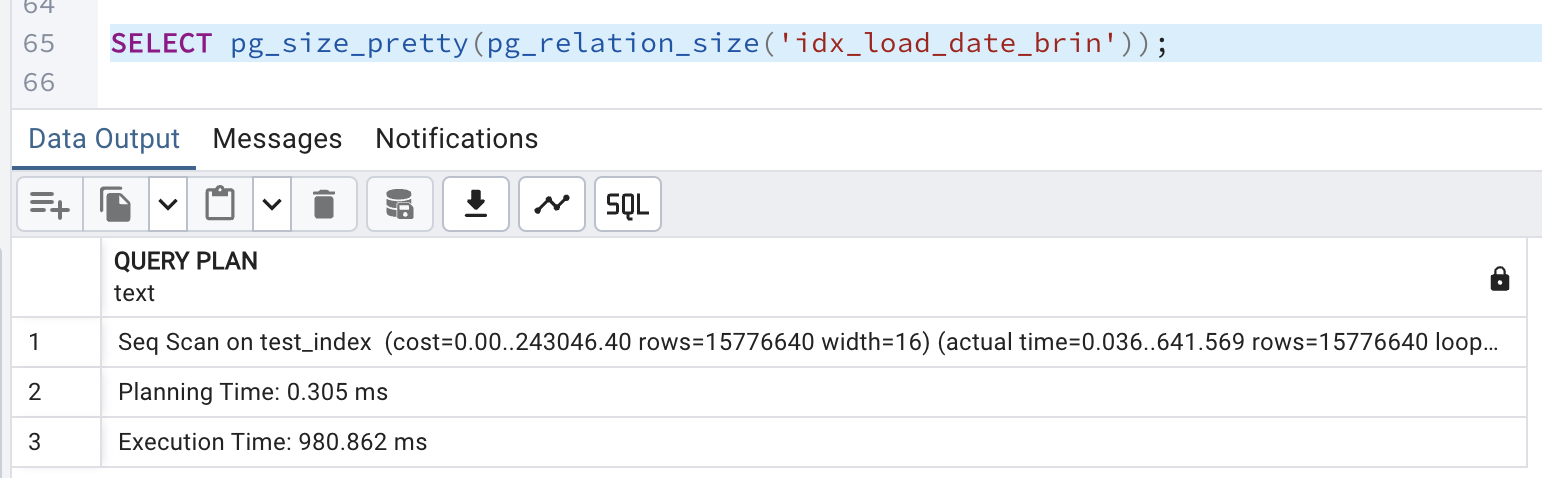


It took the same time using btree index.

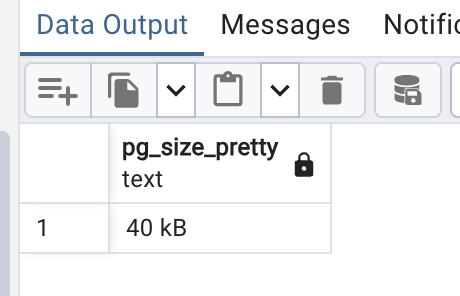


This is the size of the btree, twice less

How much time it takes for brin index. Much less, twice less

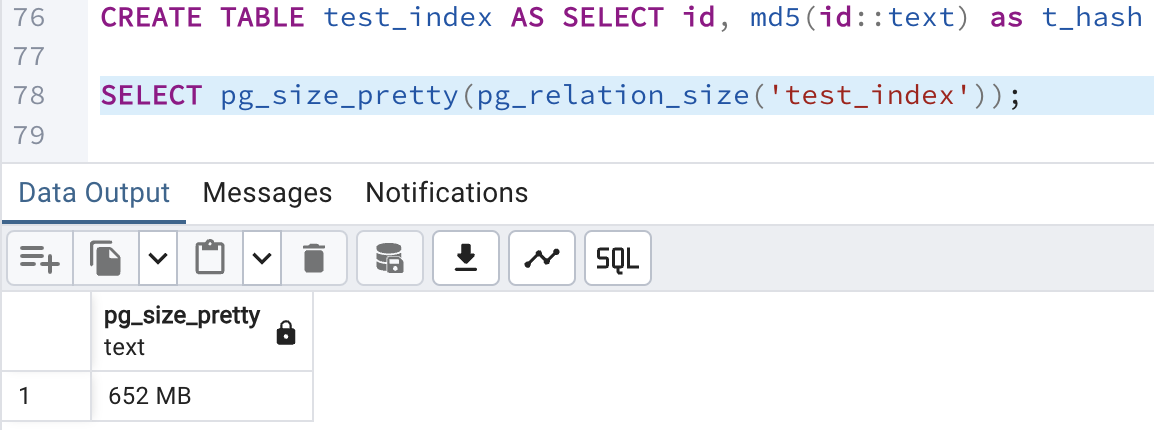


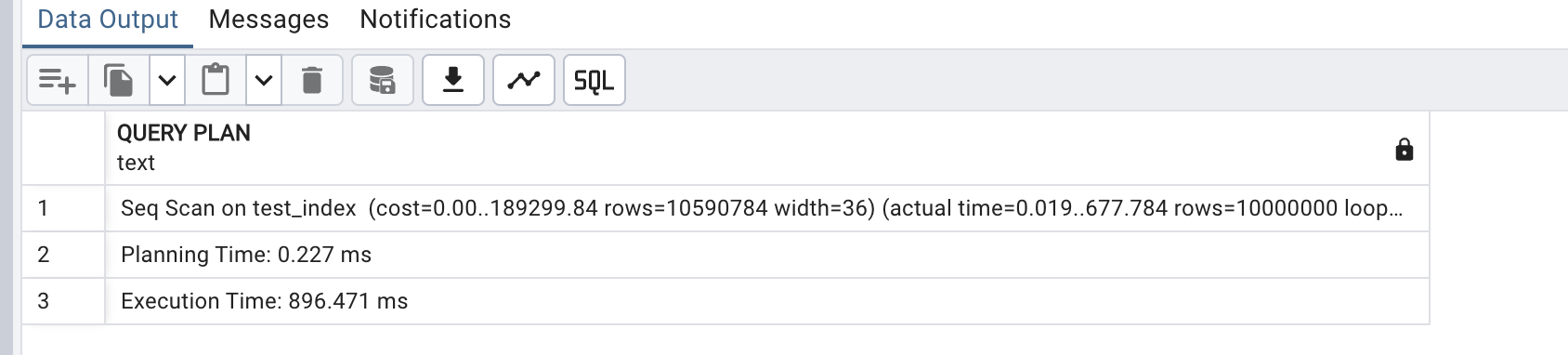
Its very small



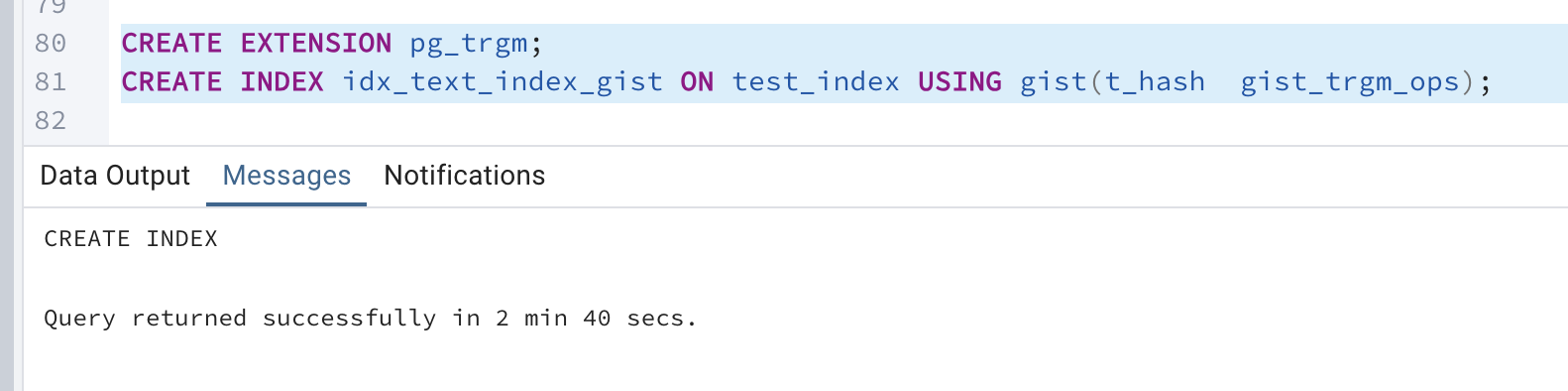
3.2

Size of the table

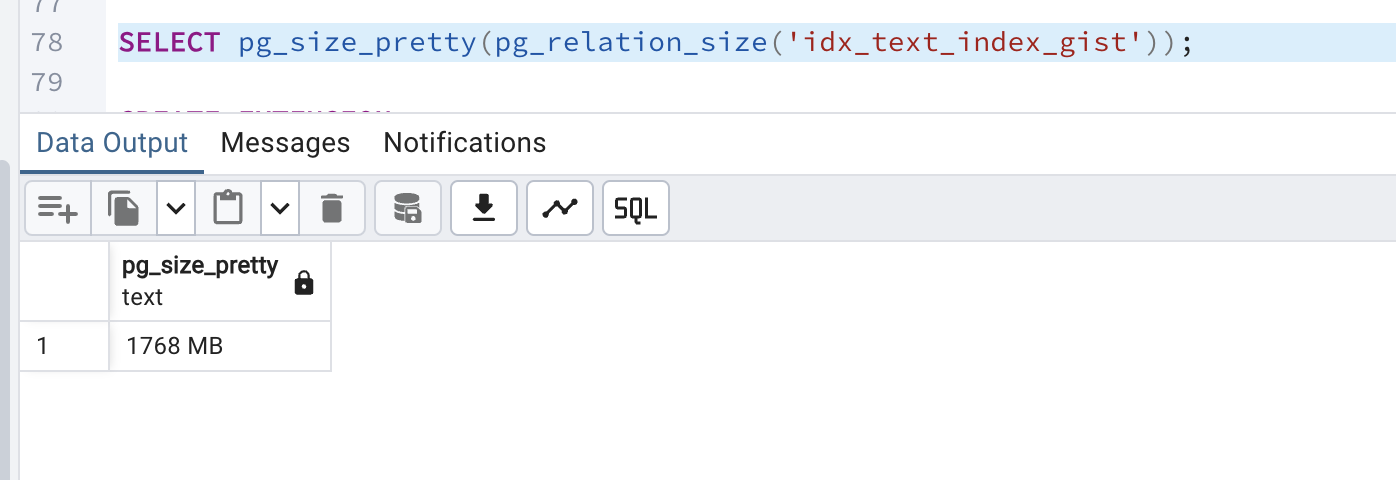


How much time it takes

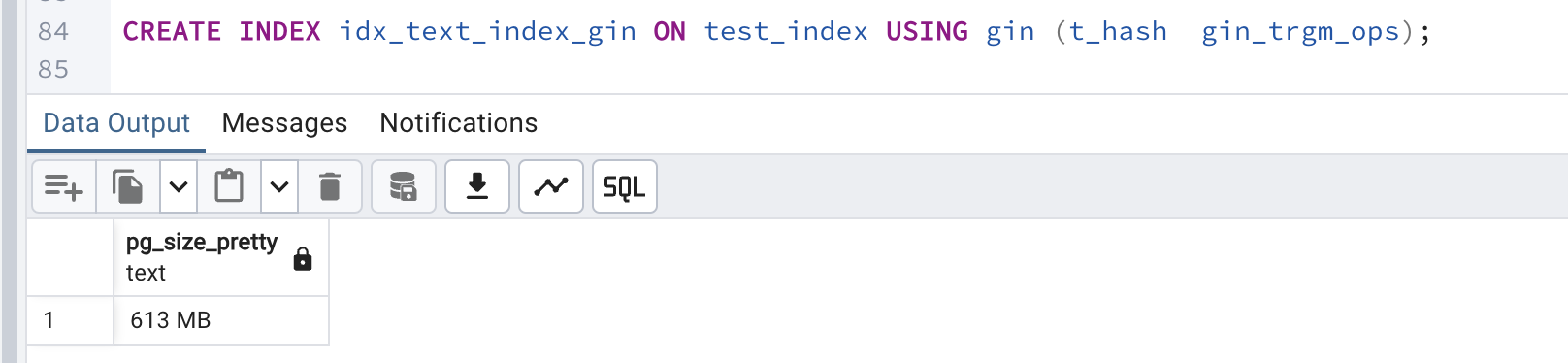
It took 2.40 minutes for gist index



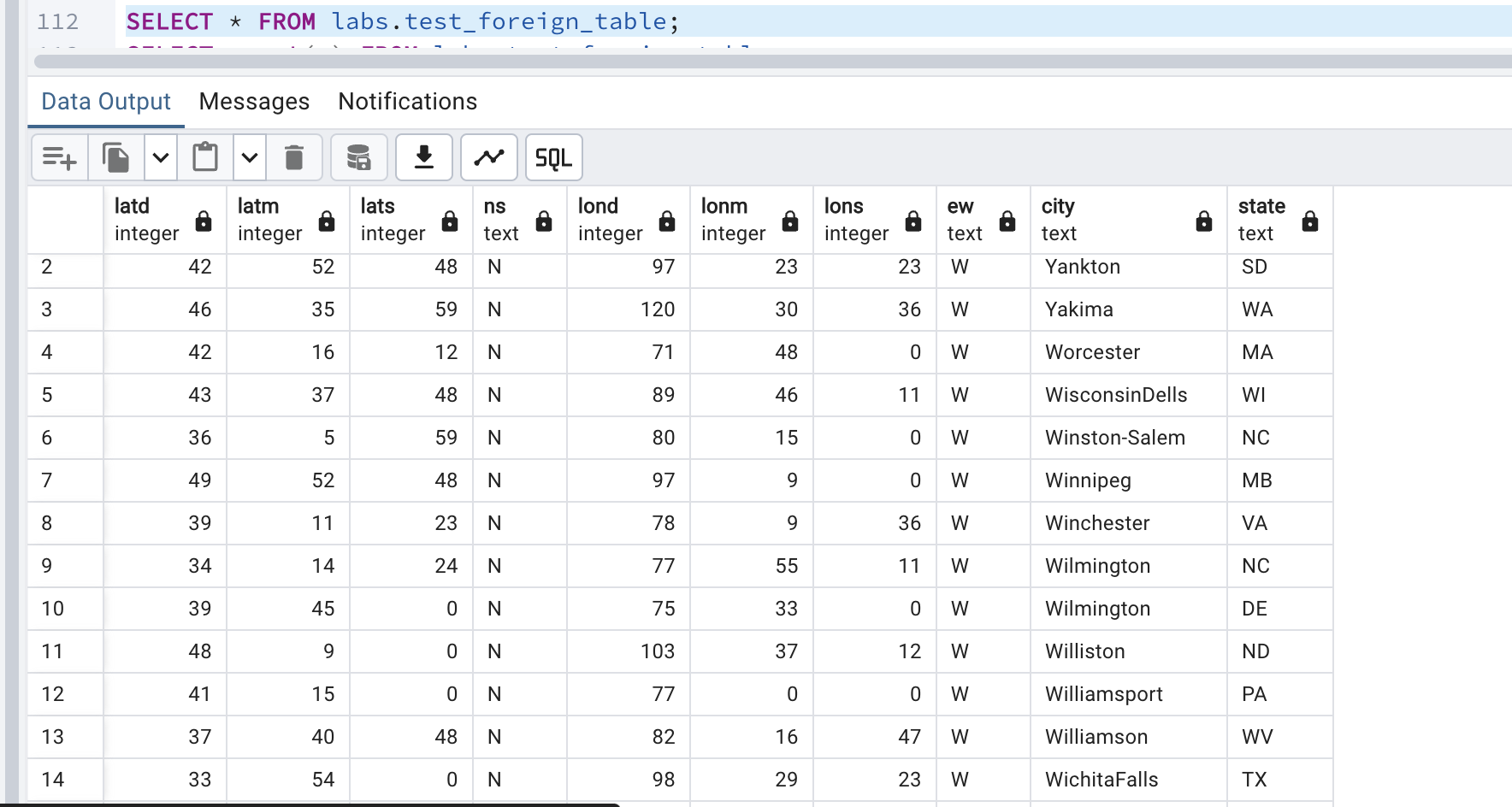
It is size of gist

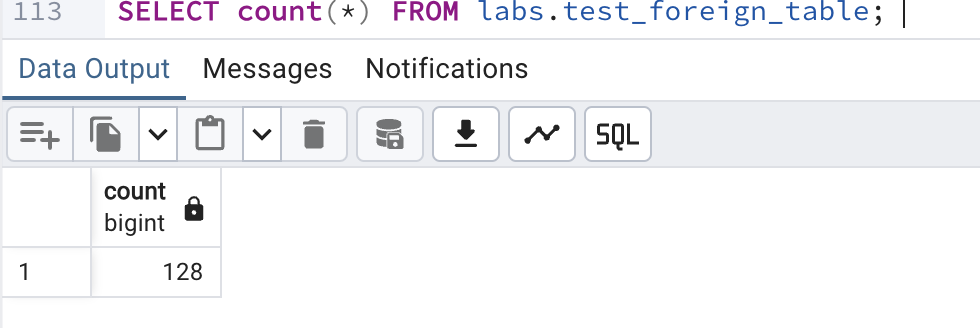


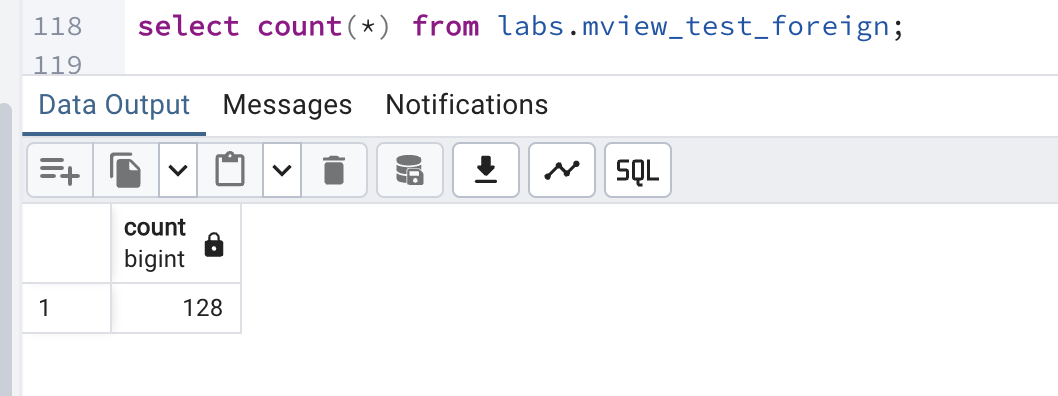
It is smaller than gist



4.0







I deleted 2 rows from the file, but the count in the materialized view remains the same because it stores a snapshot of the previous data. This means the materialized view copies the data at the time of creation and does not update automatically.