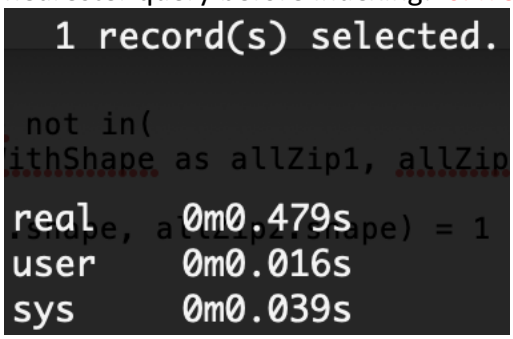
1f.
- I drop table cse532.facilitycertification in the beginning of the file. If no table is found, an error will print in the terminal.
- You must place the "Health_Facility_Certification_Information.csv" under the same directory as "**createfacilititycertificationtable.sql**" to load data successfully.

1g. I drop indexes in the beginning of the "**createindexes.sql**" file. If no indexes with the same name are found, errors will print in the terminal.

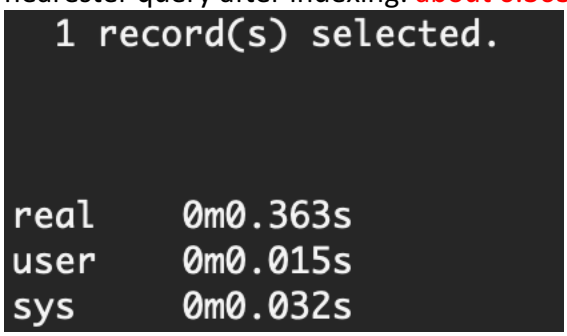
3. "**noerzips.sql**"
- I took the substring for the zip codes that have more than five digits.
  For example: 12345-5678 becomes 12345
- I also only look at the zip codes that are both in the Facility table and the USZip table.
  Which there is a total of 773 unique valid zip codes.

4. Indexes: I use the command "time db2 -vtf noerzips.sql" to estimate the time took to run each query.

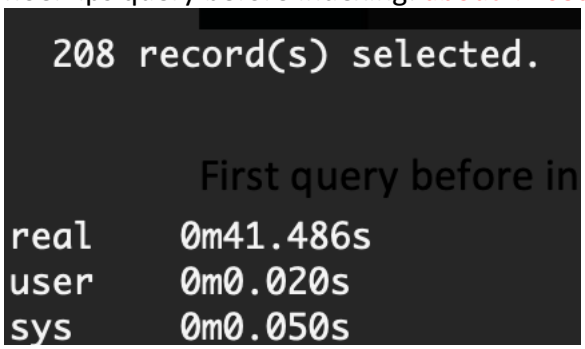
nearester query before indexing: 0.479 seconds



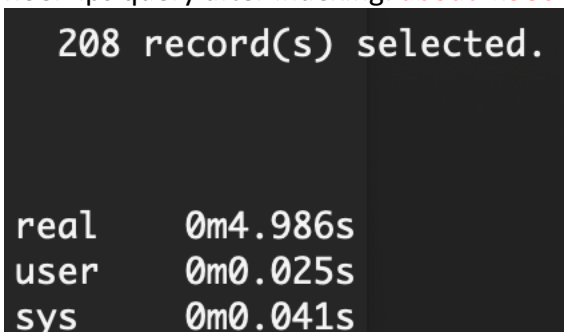
nearester query after indexing: about 0.363 seconds



noerzips query before indexing: about 41 seconds



noerzips query after indexing: about 4.986 seconds



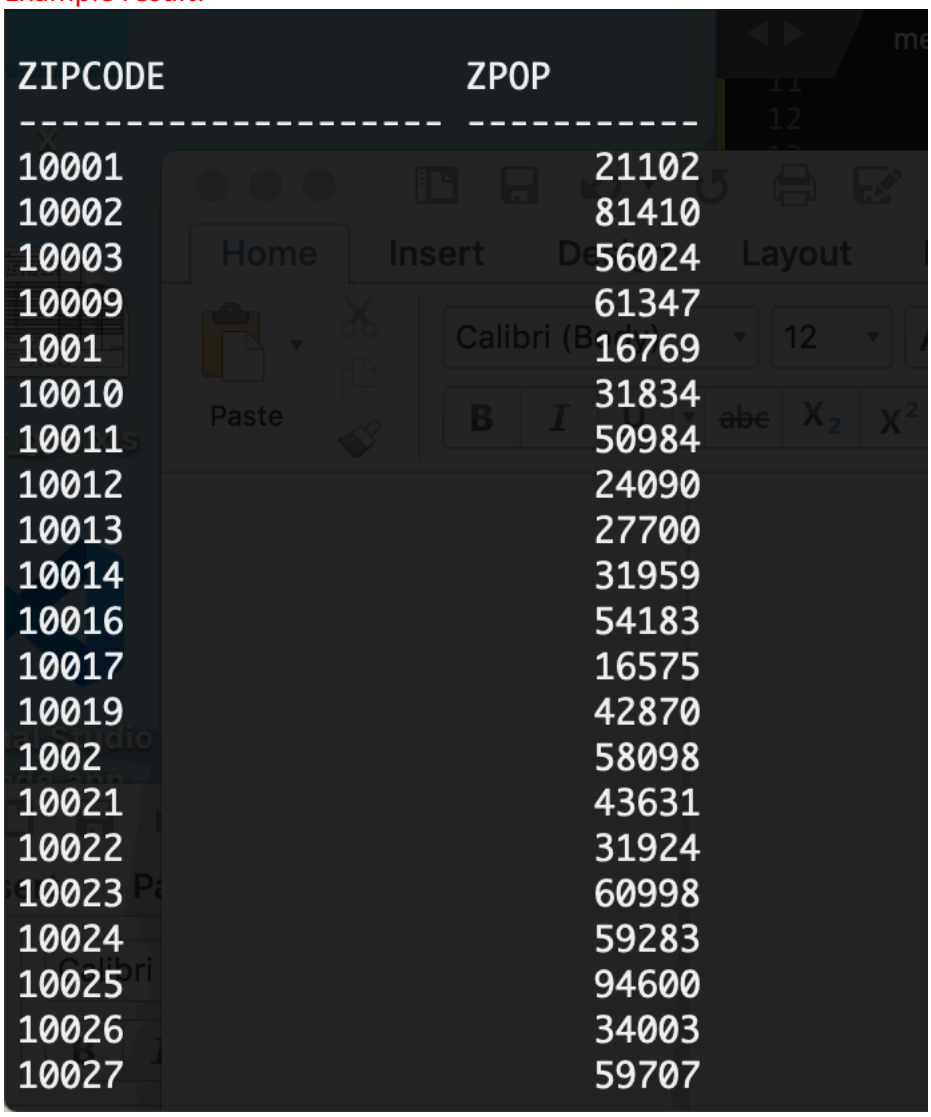
5. There are two files I wrote for this part: **mergezip1.sql** and **mergezip.sql**

- **Mergezip1.sql** creates the "cse532.neighborRelationship" table and the "cse532.resultTable" table. I used "ST_Intersects" to find all neighbors for the zipcodes that have population less than the average population (which is 12,216) and store the relationship in the first table. Then, I stored all zipcodes that have population greater than 12,216 in the second table.

- **Mergezip.sql** creates a stored procedure that goes through the "cse532.neighborRelationship" table one by one using Log(n) runtime to sum up all the zipcode population around a target zip code (the zip code that had population less than avg) and insert a new row in the "cse532.resultTable".

To run the query: (The first query takes around 10 minutes, third takes around 4 minutes)
1. db2 -tf mergezip1.sql
2. db2 -td@ -f mergezip.sql
3. db2 call "merge_zip(?)"
4. db2 select * from cse532.resultTable

Example result:



ZIPCODE	ZPOP
10001	21102
10002	81410
10003	56024
10009	61347
1001	16769
10010	31834
10011	50984
10012	24090
10013	27700
10014	31959
10016	54183
10017	16575
10019	42870
1002	58098
10021	43631
10022	31924
10023	60998
10024	59283
10025	94600
10026	34003
10027	59707