Nearest Elementary, Middle, and Highschool in PgAdmin/QGIS

Timothy Elvira

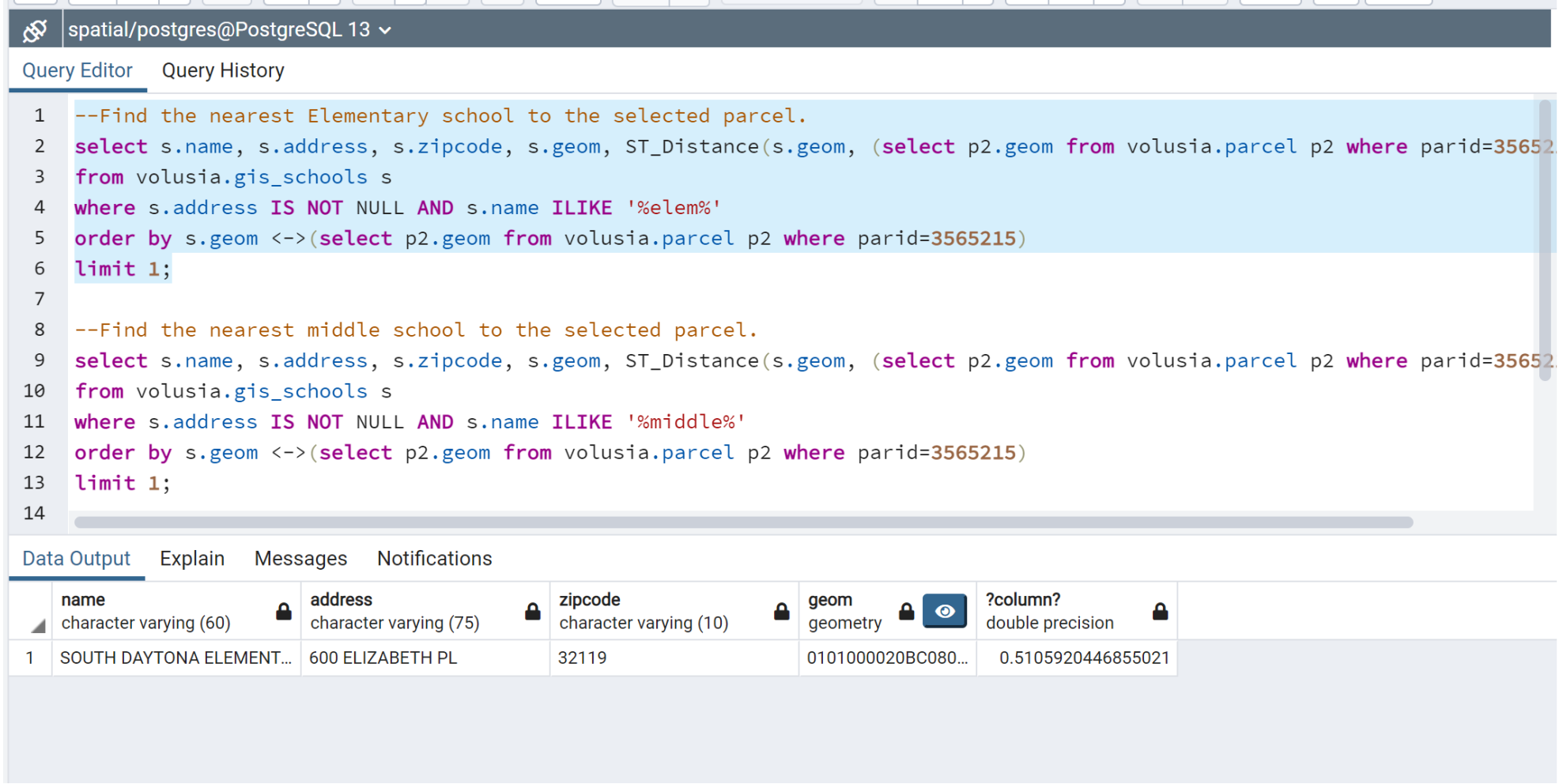
CS540

Appending to Volusia.Parcels

- ALTER TABLE volusia.parcel
- add column nearest_Elem_School VARCHAR,
- add column distance_To_Elem_School double precision,
- add column nearest_Middle_School VARCHAR,
- add column distance_To_Middle_School double precision,
- add column nearest_High_School VARCHAR,
- add column distance_To_High_School double precision;

This is for the columns to add the data in.

- Its important to use get_gis_schools. Bat file in order to grab the shape files and use them in PGAdmin.



Example queries to get the knn of the closes elementary school. I grab the name, address, zipcode, geoms, and distance between the school and the parcel 356215.

- --Find the nearest Elementary school to the selected parcel.

```
select s.name, s.address, s.zipcode, s.geom, ST_Distance(s.geom, (select p2.geom from volusia.parcel p2 where parid=3565215))/5280
from volusia.gis_schools s
where s.address IS NOT NULL AND s.name ILIKE '%elem%'
order by s.geom <->(select p2.geom from volusia.parcel p2 where parid=3565215)
limit 1;
```

- --Find the nearest middle school to the selected parcel.

```
select s.name, s.address, s.zipcode, s.geom, ST_Distance(s.geom, (select p2.geom from volusia.parcel p2 where parid=3565215))/5280
from volusia.gis_schools s
where s.address IS NOT NULL AND s.name ILIKE '%middle%'
order by s.geom <->(select p2.geom from volusia.parcel p2 where parid=3565215)
limit 1;
```

- --find the nearest high school to the selected parcel.

```
select s.name, s.address, s.zipcode, s.geom, ST_Distance(s.geom, (select p2.geom from volusia.parcel p2 where parid=3565215))/5280
from volusia.gis_schools s
where s.address IS NOT NULL AND s.name ILIKE '%high%'
order by s.geom <->(select p2.geom from volusia.parcel p2 where parid=3565215)
limit 1;
```

The hard part… Looping in PGAdmin

- First step:

Ensure that the columns are Null Using:

Update volusia.parcelset nearest_elem_school = null, distance_To_Elem_School = null;

This is because we're going to use the fact that they are Null to loop through them.

I provided the SQL files to do all these loops: elem, middle, and highschool and they each take about 10 minutes to run.

Declaration

```
DO
LANGUAGE plpgsql
$$
DECLARE
g1 geometry;
rec RECORD;
es VARCHAR(45);
distanceFromES float;
```

This section just declares the variables within the loop. IN the case of Rec, that's just each individual parcel within the volusia.parcels. Es and distanceFromES are placeholder variables that we're going to use to update the volusia.parcels columns.

```
BEGIN
    for rec in select parid, geom from volusia.parcel
        where nearest_Elem_School is NULL AND distance_To_Elem_School is NULL and geom IS NOT NULL loop
        g1:=rec.geom;

        select into es s.name
            from volusia.gis_schools s
            where s.address IS NOT NULL AND s.name ILIKE '%elem%' AND s.theme = 'PUBLIC'
            order by s.geom <->(g1)
            limit 1;


        select into distanceFromES ST_Distance(s.geom, (g1))/5280 as distance
            from volusia.gis_schools s
            where s.address IS NOT NULL AND s.name ILIKE '%elem%' AND s.theme = 'PUBLIC'
            order by s.geom <->(g1)
            limit 1;

        update volusia.parcel set nearest_Elem_School = es, distance_To_Elem_School = distanceFromES where parid=rec.p
        RAISE NOTICE 'set to % % %', rec.parid, es, distanceFromES;
    END LOOP;
End;

$$;
```

The loop

- The previous slide is the loop query. The for statement loops through each parcel (rec) that has null values for the elem name and distance and also has valid geoms. G1 is set to be the current rec's geoms.

- Then the s.name is selected to be the es variable using the closest school near g1.

- Distance is selected to be inputted into the distanceFromES using the closest elementary school near g1.

The loop, the update

- We update the volusia.parcels columns using:

```
update volusia.parcel set nearest_Elem_School = es,
distance_To_Elem_School = distanceFromES where parid=rec.parid ;
        RAISE NOTICE 'set to % % %', rec.parid, es, distanceFromES;
```

The raise notice is not necessarily needed because you're updating 300k parcels but you can show them if you want. Not recommended but it also didn't slow me down.

Verification

- Repeat for the Middle School and Highschool SQL files in your PGAdmin

- Once that is done, query your volusia.parcels with:

```
select geom, parid, nearest_elem_school, distance_to_elem_school,
nearest_middle_school, distance_to_middle_school,
nearest_high_school, distance_to_high_school from volusia.parcel
where nearest_elem_school IS NOT NULL AND nearest_middle_school
IS NOT NULL AND nearest_high_school IS NOT NULL
limit 50;
```

- Query and Select the Columns to View the data.

spatial/postgres@PostgreSQL 13

Query Editor | Query History

```
1  select geom, parid, nearest_elem_school, distance_to_elem_school, nearest_middle_school, distance_to_middle_school, near
2  where nearest_elem_school IS NOT NULL AND nearest_middle_school IS NOT NULL AND nearest_high_school IS NOT NULL
3  limit 50;
4
```

Data Output | Explain | Messages | Notifications

	geom geometry	parid double precision	nearest_elem_school character varying	distance_to_elem_school double precision	nearest_middle_school character varying	distance_to_middle_school double precision	n c
1	0106000020BC080...	3022629	ORMOND BEACH ELEMENT...	2.484717751516029	ORMOND BEACH MIDDLE SC...	1.0697864413143061	S
2	0106000020BC080...	3023561	ORMOND BEACH ELEMENT...	2.427454254908564	ORMOND BEACH MIDDLE SC...	1.065666881566863	S
3	0106000020BC080...	6335133	HORIZON ELEMENTARY SC...	0.9943931071845498	SILVER SANDS MIDDLE SCHO...	2.3282008717613785	S
4	0106000020BC080...	3025598	OSCEOLA ELEMENTARY SC...	2.8334219304949957	ORMOND BEACH MIDDLE SC...	1.4586633766482053	S
5	0106000020BC080...	3025601	OSCEOLA ELEMENTARY SC...	2.829415637060652	ORMOND BEACH MIDDLE SC...	1.4646479737223697	S
6	0106000020BC080...	3026187	ORMOND BEACH ELEMENT...	2.79610662491628	ORMOND BEACH MIDDLE SC...	1.3839237642965345	S
7	0106000020BC080...	3028104	ORMOND BEACH ELEMENT...	3.122750945438913	ORMOND BEACH MIDDLE SC...	1.5483106748844944	S
8	0106000020BC080...	3028112	ORMOND BEACH ELEMENT...	3.1241490537585928	ORMOND BEACH MIDDLE SC...	1.5427650749887938	S

PostGis Layer

- In QGIS, add a postgis layer and select the volusia.parcels table.
- Go to Symbology and select nearest_elem/middle/high_school and click classify, this should classify each parcel with their respective nearest school.

View on QGIS

- Bit of a mess but those parcels are match with those highschools. Add the appropriate categorization in QGIS to see elementary and middle respectively. I just used the categorization button and all the school names were used.