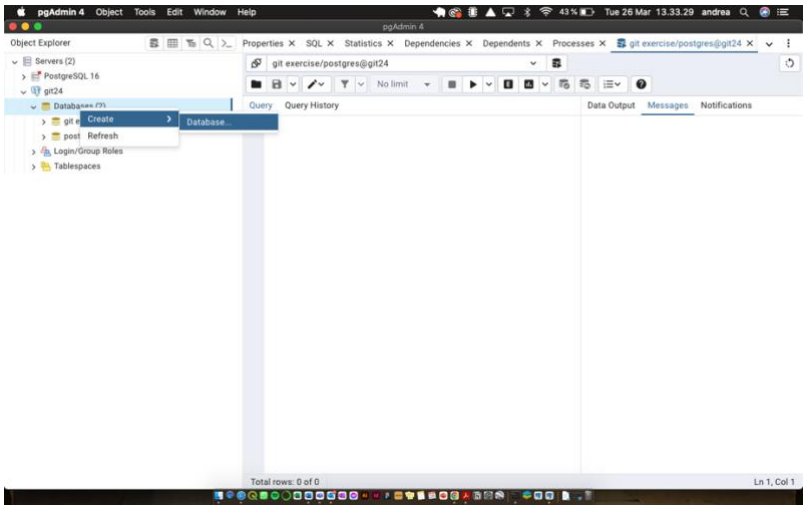
GeoInformationTechnology – DataBase Exercise – Andrea Mantegna

Task 1. The first step is to create a new spatial database using PGAdmin.

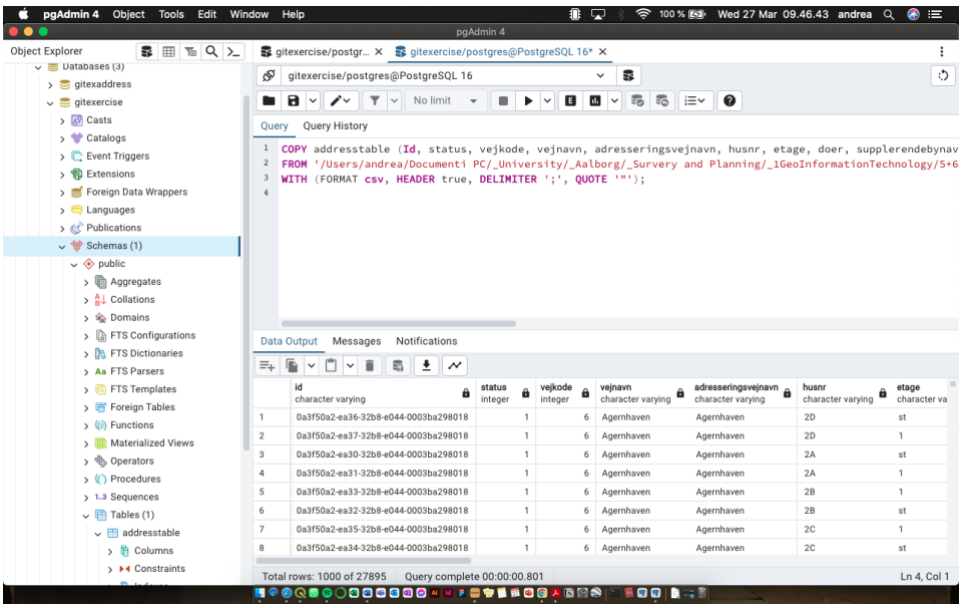


Task 2. Now we need to create a table *in* the database **addresstable**. (The screenshot refers to a previous attempt to create a database therefor the name could be different, but the process is the same.)

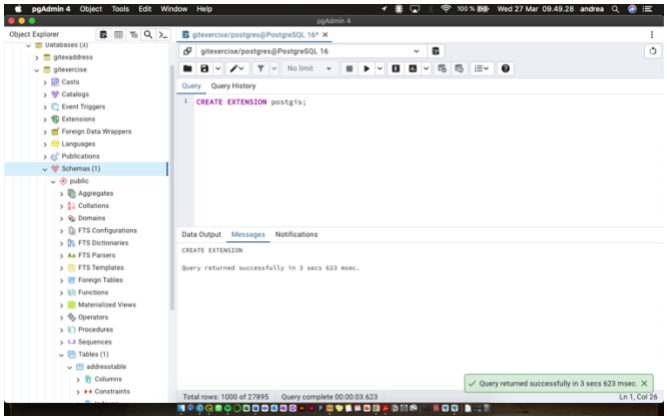
```
CREATE TABLE tableaddresses (
id VARCHAR,
status INTEGER,
vejkode INTEGER,
vejnavn VARCHAR,
adresseringsvejnavn VARCHAR,
"husnr" VARCHAR,
etage VARCHAR,
doer VARCHAR,
supplerendebynavn VARCHAR,
postnr INTEGER,
postnrnavn VARCHAR,
kommunekode INTEGER,
kommunenavn VARCHAR,
ejerlavkode FLOAT,
ejerlavnavn VARCHAR,
matrikelnr VARCHAR,
esrejendomsnr FLOAT,
etrs89koordinat_oest FLOAT,
etrs89koordinat_nord FLOAT,
regionskode INTEGER,
regionsnavn VARCHAR,
jordstykke_ejerlavnavn VARCHAR
);
```

To create the table, we have to study each column of the CSV file “addresses.csv”. Each column would then be categorized at **varchar, integer** or **float**.

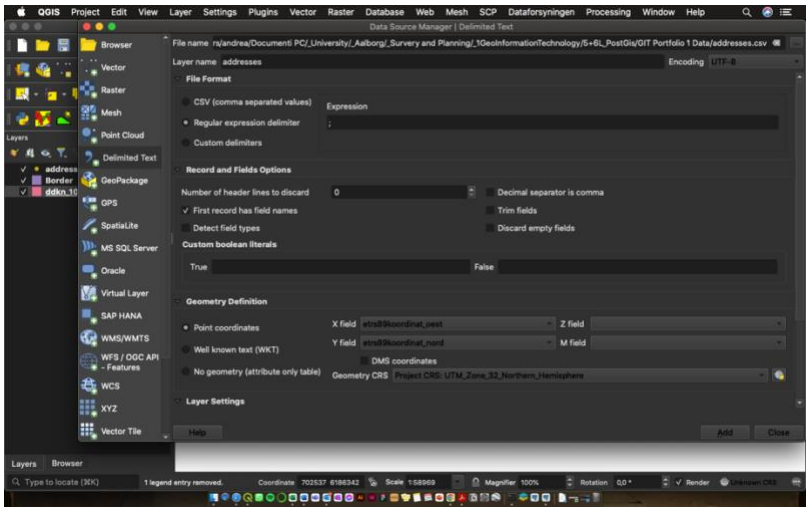
After the table is has been created, the command COPY allows you to connect the CSV file to the table.

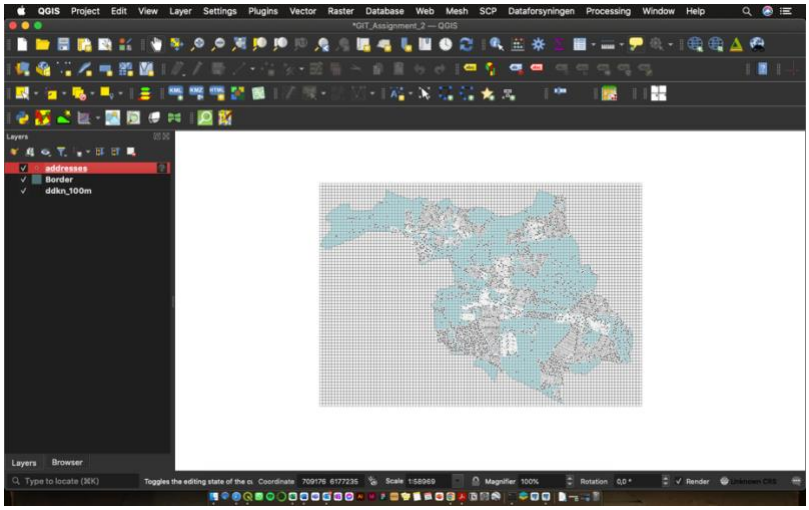


Task 3. Before loading the grid dataset on Qgis we create an extension on PGAdmin to connect it to Qgis.



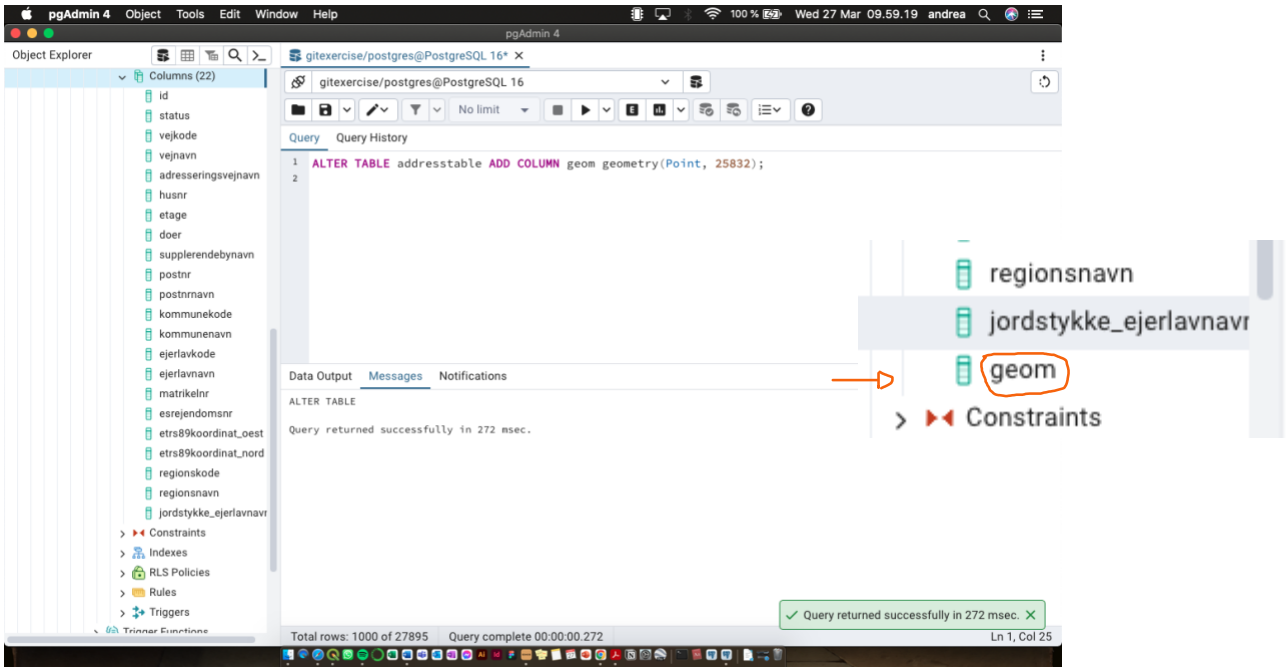
Task 4. Using **Qgis** we load the grid dataset CSV file through **Layer > Add layer > Add Delimited Text Layer.** We set the " ; " to be the delimiter character, the **X and Y field** with the coordinate from the file "**etrs89koordinat_oest**" and ""**etrs89koordinat_nord**" and lastly the reference system to **Project CRS: UTM_Zone_32_Northern_Hempishere.**

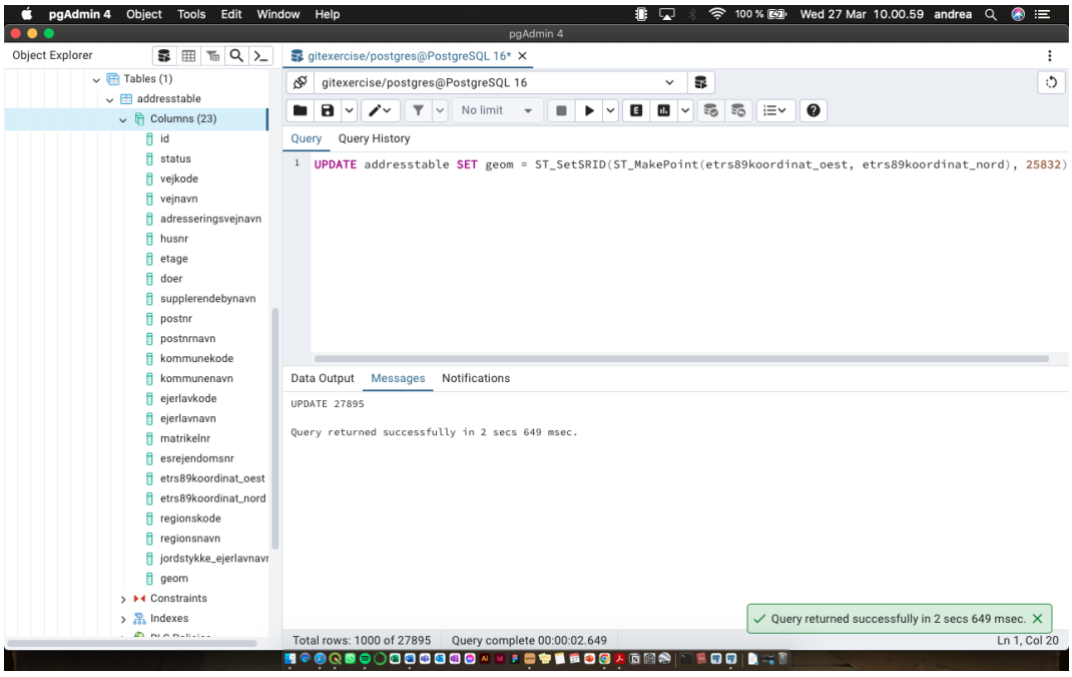




Task 5. Using PGAdmin we add a geometry column to the address dataset and then populate it with spatial information of the geometry from the X and Y coordinate.

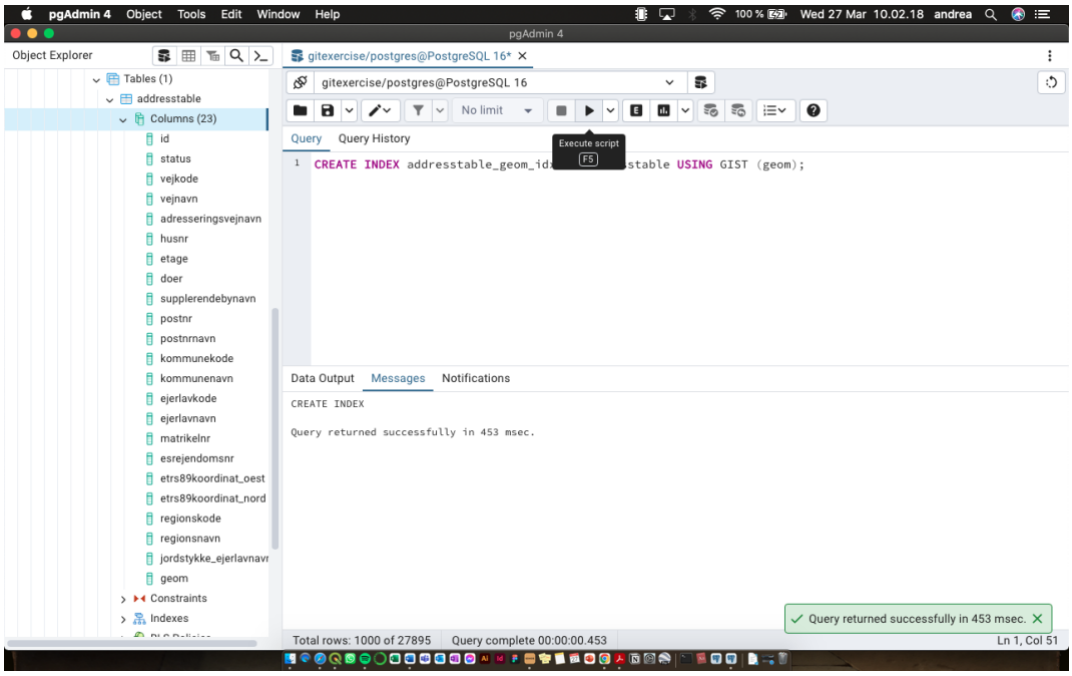
ALTER TABLE addresstable ADD COLUMN geom geometry(Point, 25832);





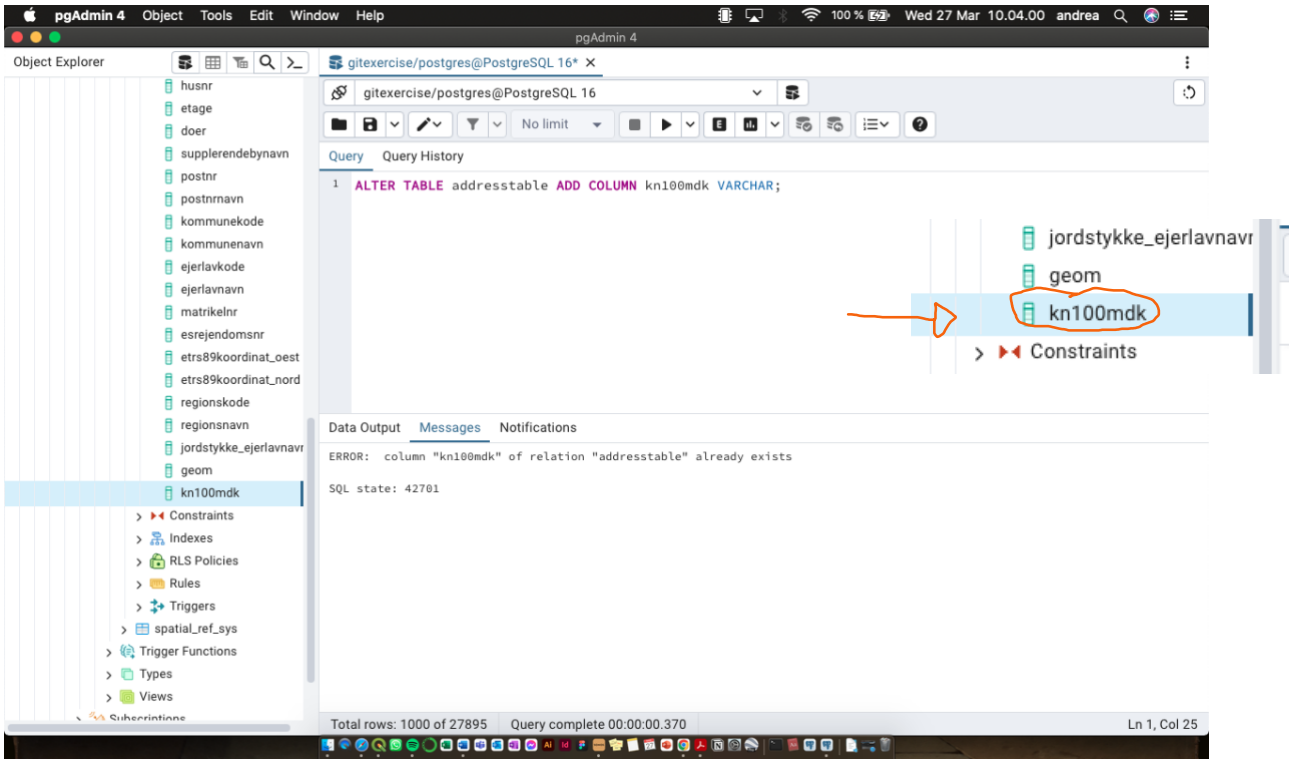
UPDATE tableaddresses SET geom = ST_SetSRID(ST_MakePoint(etrs89koordinat_oest, etrs89koordinat_nord), 25832);

Task 6. We then crate a spatial index for the column with the addresses geometry.

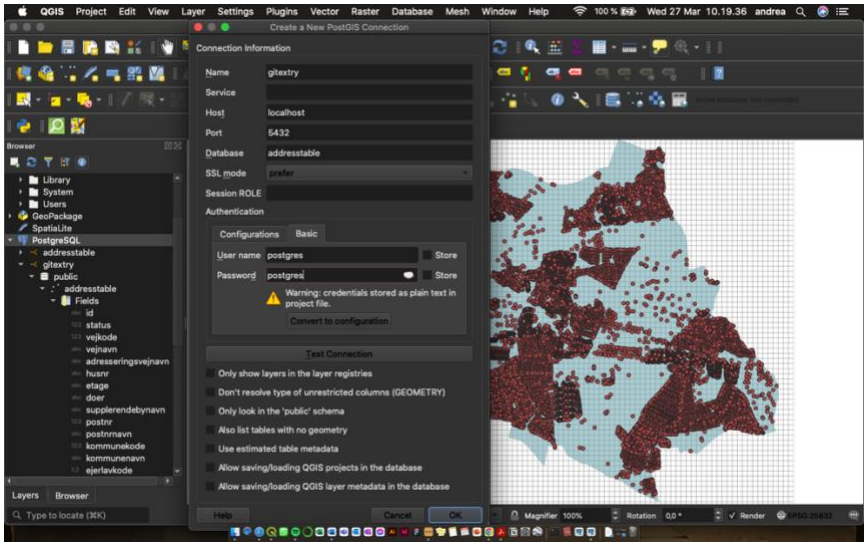


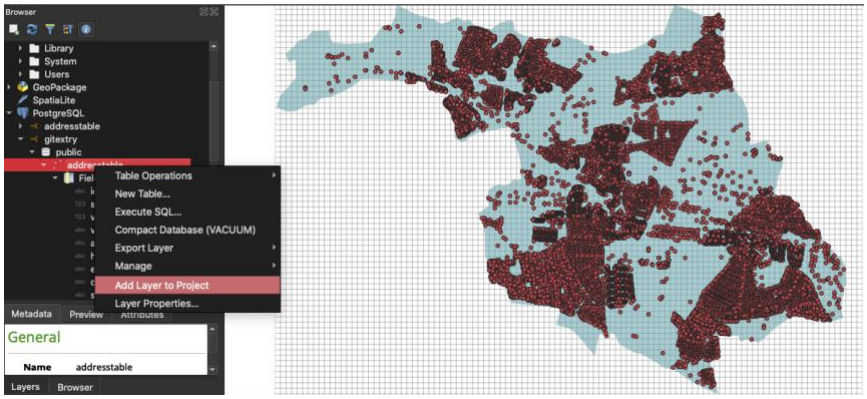
CREATE INDEX tableaddresses_geom_idx ON tableaddresses USING GIST (geom);

Task 7. Then we add a column to the addresses table named “kn100mdk” and populate that column with the value from the grid dataset using an intersection tool on Qgis. This adds the cell id to each address.

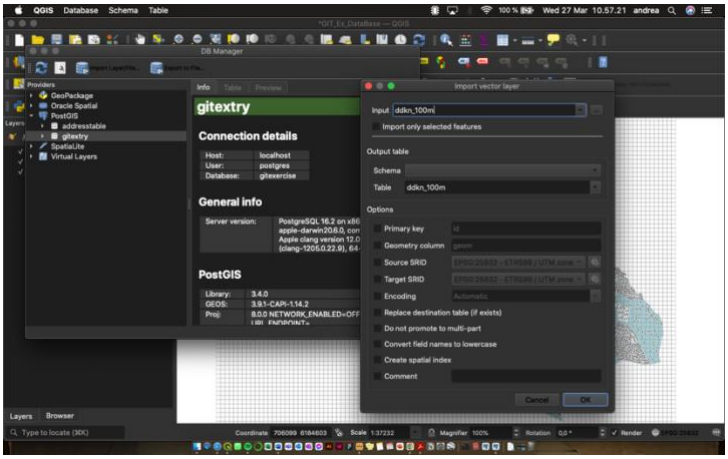


Using the plugin for PostGis in Qgis we connect the database to Postgres.

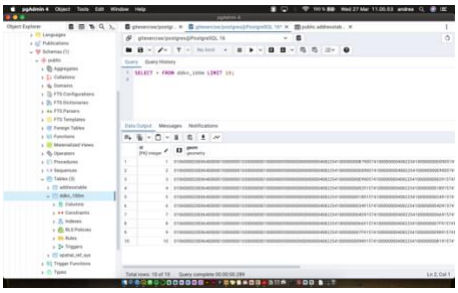




Task 8. First we import the shapefile ddkn_100m.shp in PGAdmin using Qgis in order to perform the intersection between the addresses and the grid.



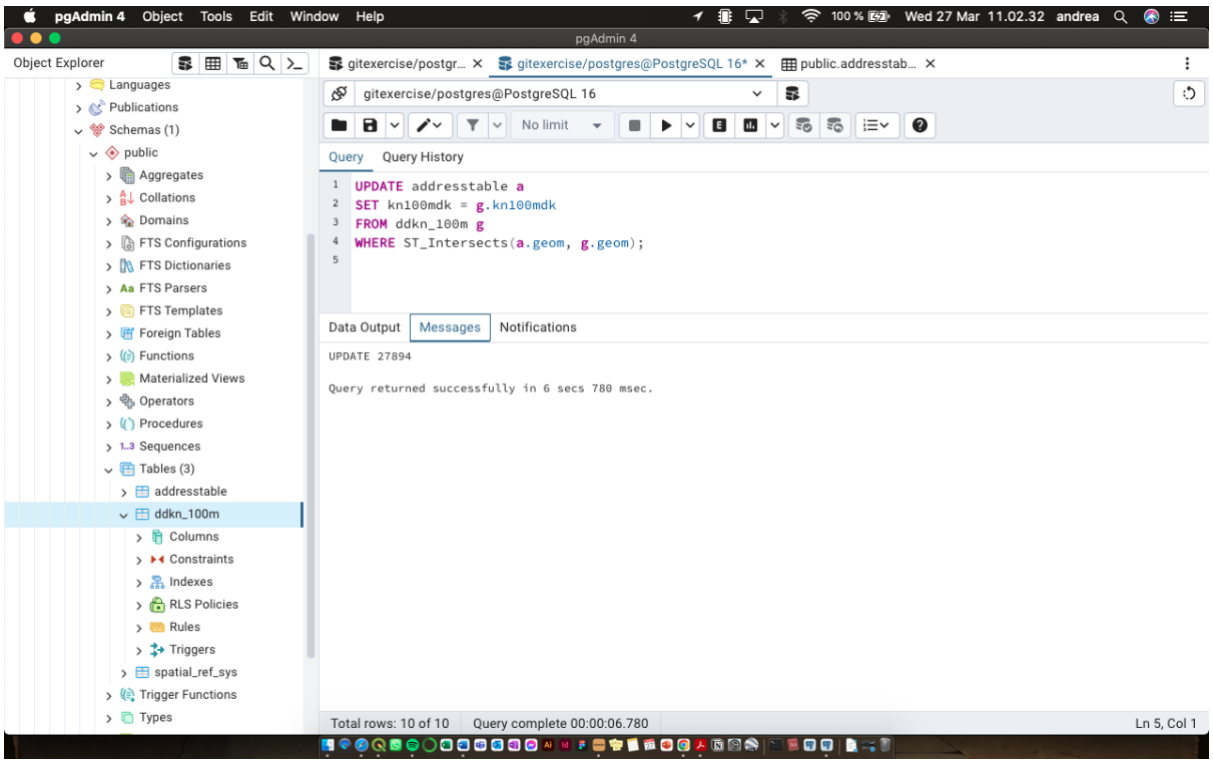
We verify that is has been imported correctly.



Then we run the code.

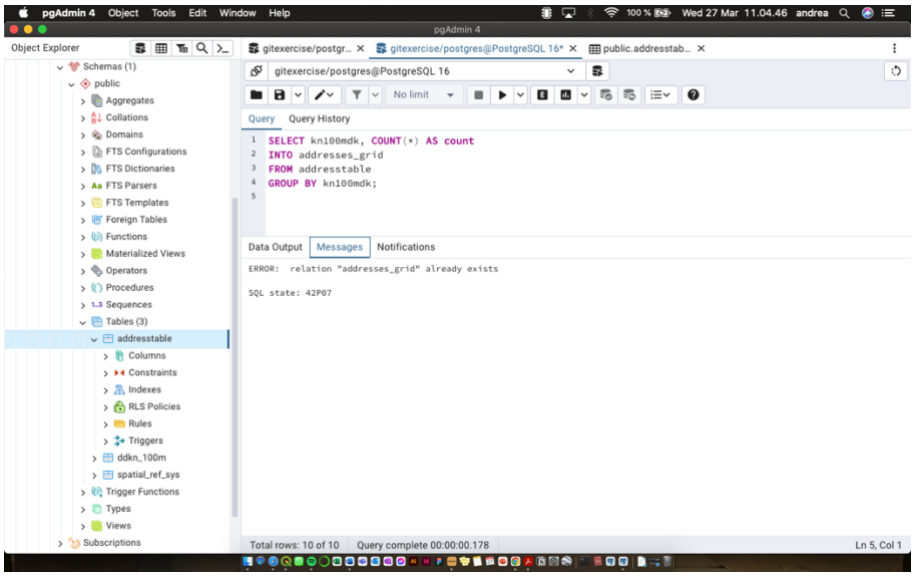
```
UPDATE addresstable AS a
```

```
SET kn100mdk = g.kn100mdk
FROM addresstable AS g
WHERE ST_Intersects(a.geom, g.geom);
```



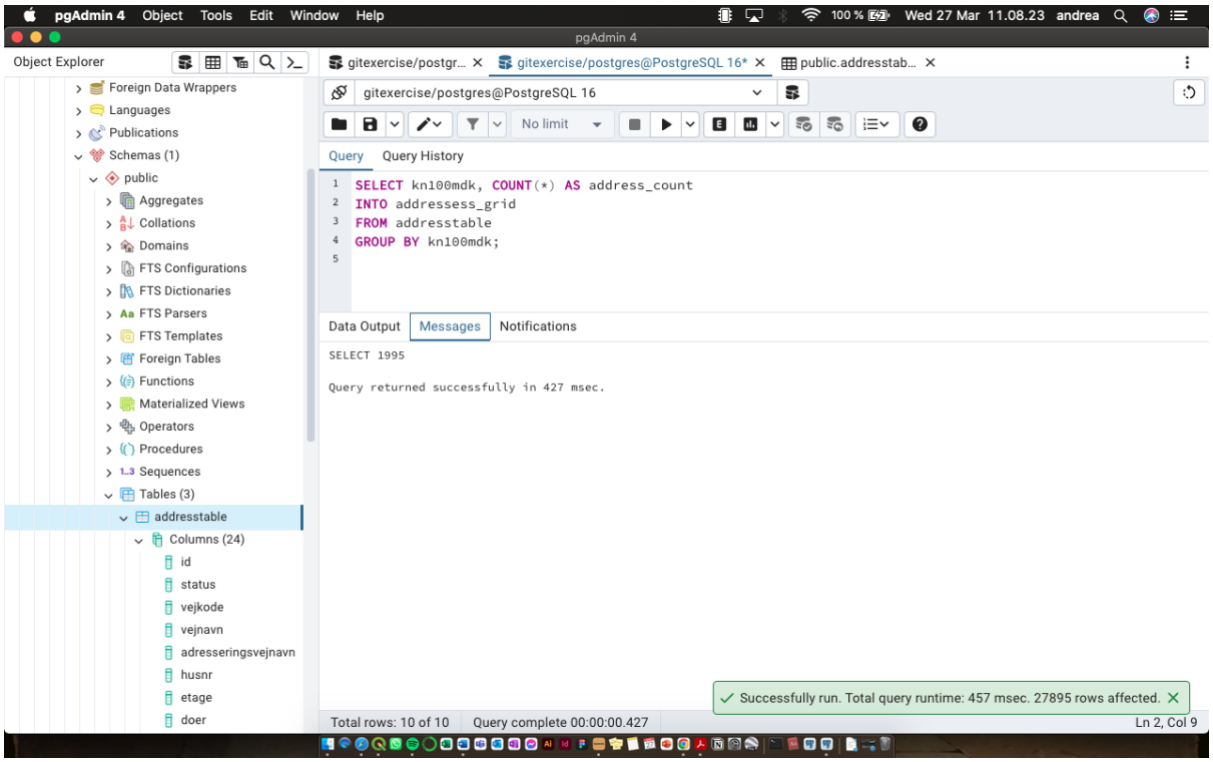
Task 9. We create a table “addresses_grid” where to establish / insert the intersection result.

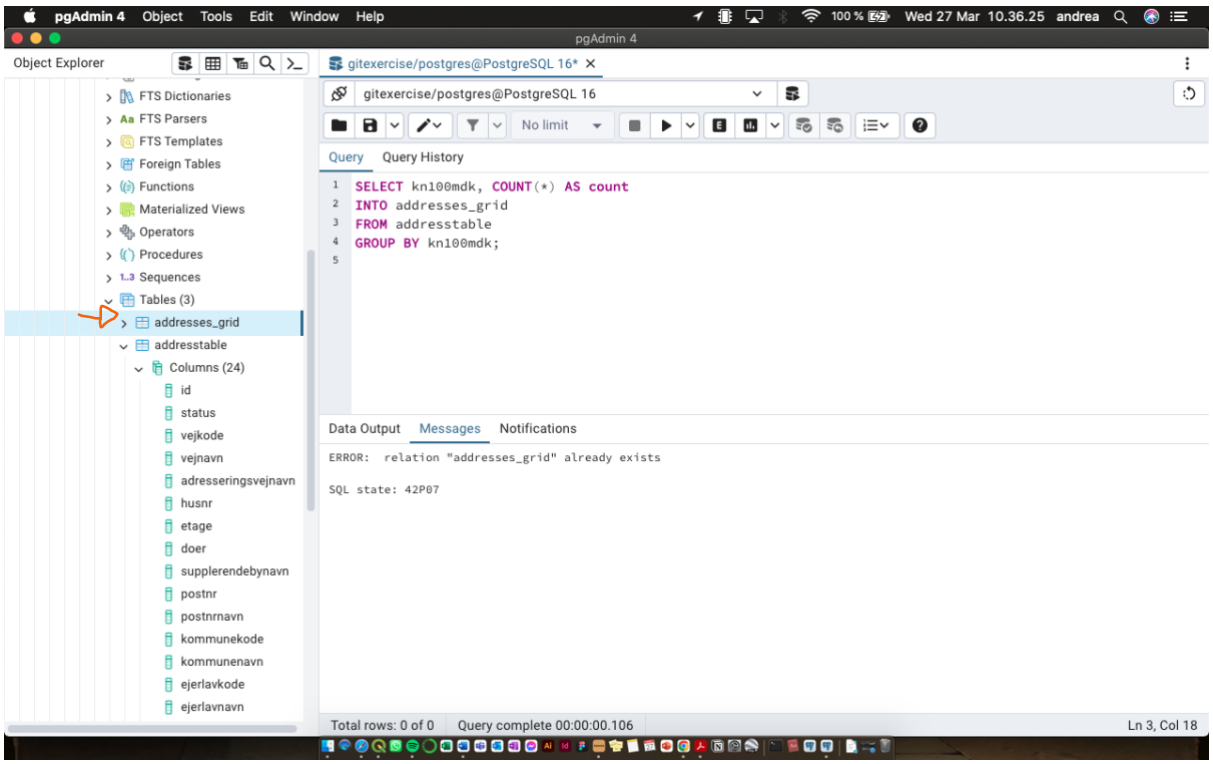
```
SELECT kn100mdk, COUNT(*) AS count
INTO addresses_grid
FROM addresstable
GROUP BY kn100mdk;
```



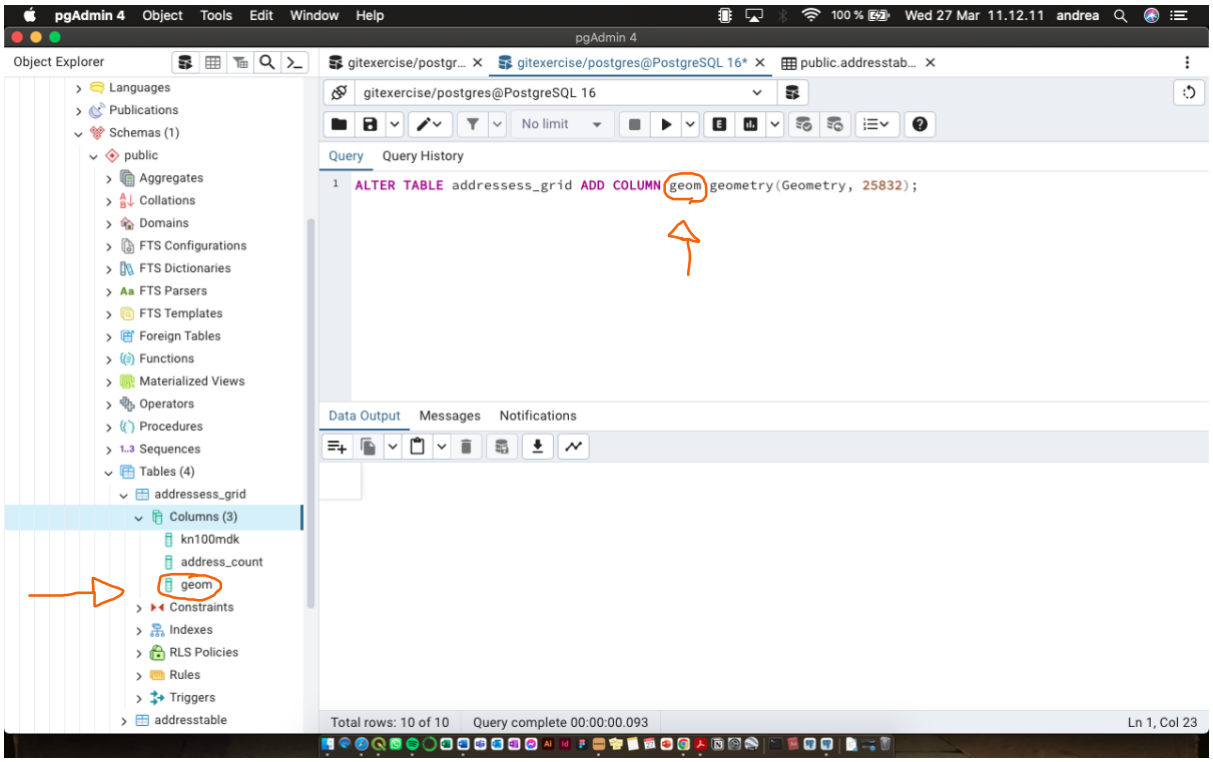
Task 10. We need to create a geometry column in the table “addresses_grid” to insert the coordinate.

```
SELECT kn100mdk, COUNT(*) AS address_count
INTO addressess_grid
FROM addresstable
GROUP BY kn100mdk;
```



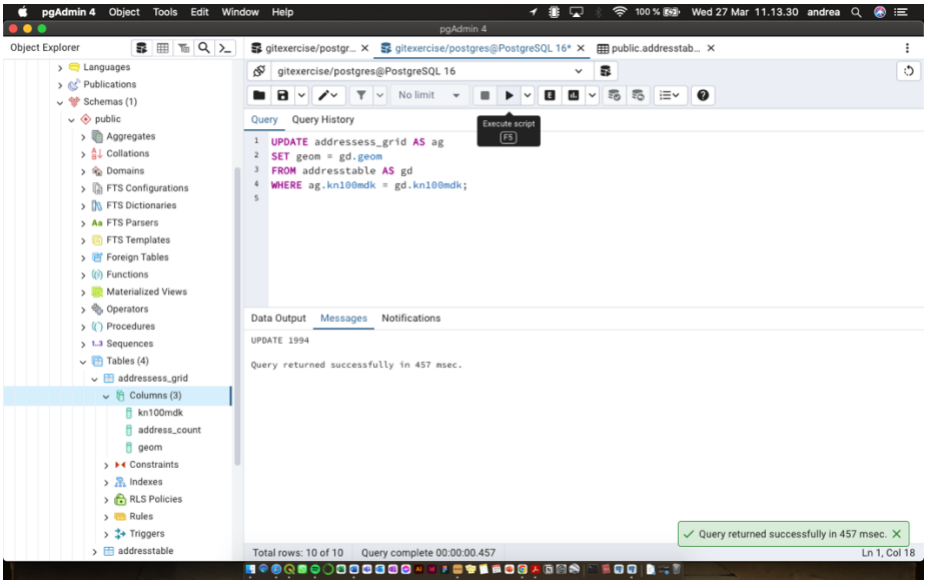


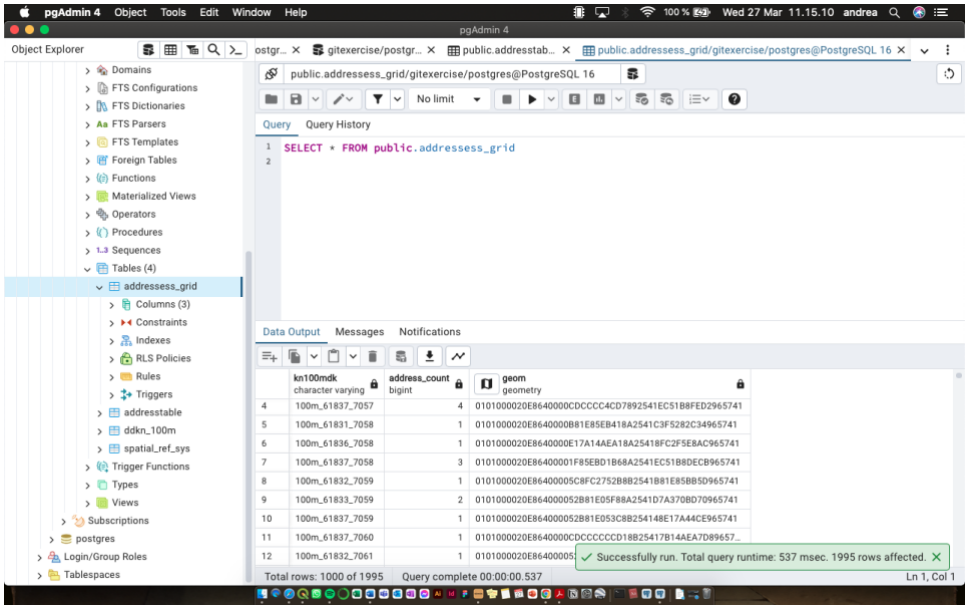
```
ALTER TABLE addresses_grid ADD COLUMN geom geometry(Geometry, 25832);
```



Then it's time to run the code that joins the geometry form the original grid dataset (ddkn_100m) to the "addresses_grid" table.

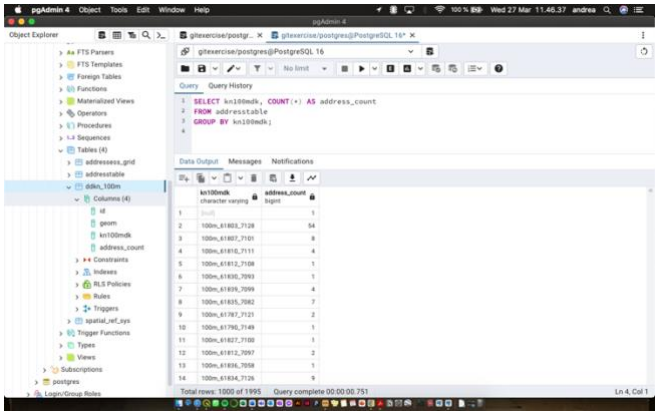
```
UPDATE addressess_grid AS ag
SET geom = gd.geom
FROM addresstable AS gd
WHERE ag.kn100mdk = gd.kn100mdk;
```

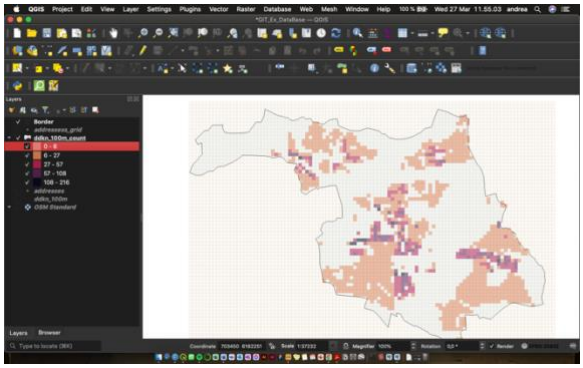




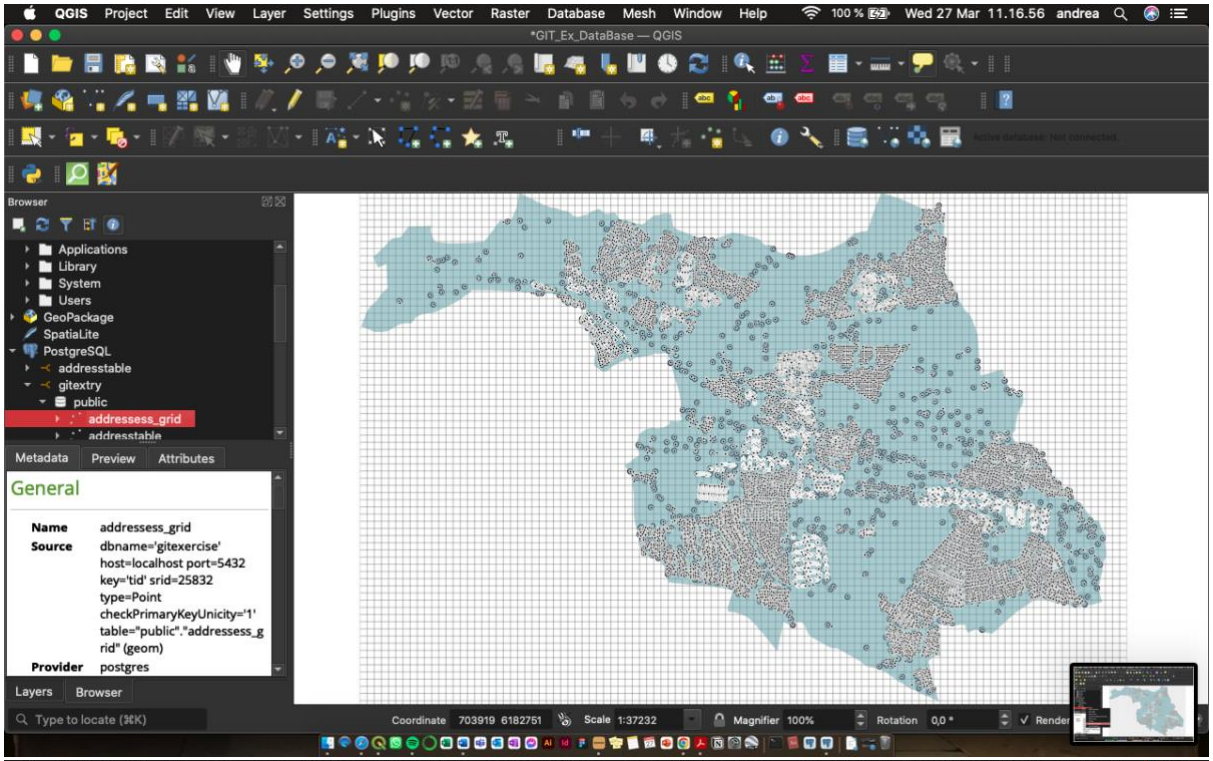
To categorize the grid and count how many addresses there are in each square we need to run this code.

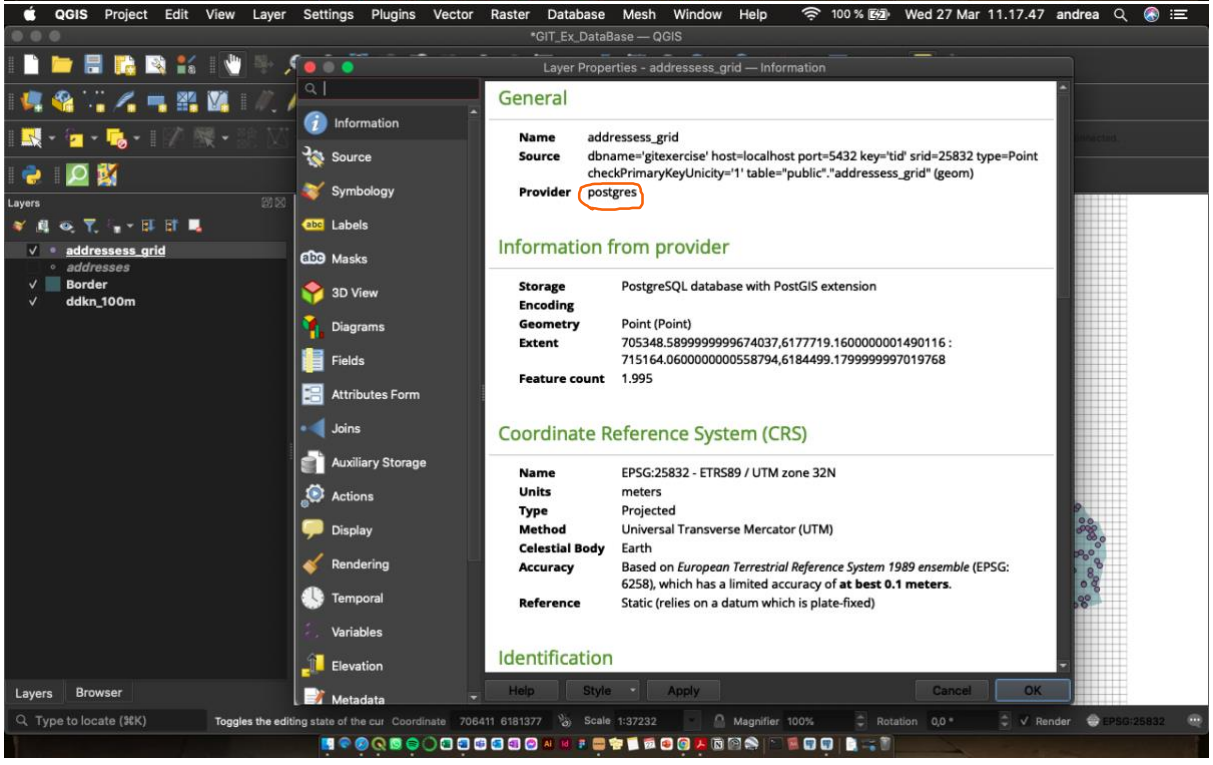
```
SELECT kn100mdk, COUNT(*) AS address_count
FROM addresstable
GROUP BY kn100mdk;
```





Task 10. The results of this code should update the spatial information on Qgis. Let's restart Qgis and visualize the results.





Task 11. Let’s add the categorized grid and enrich the map.