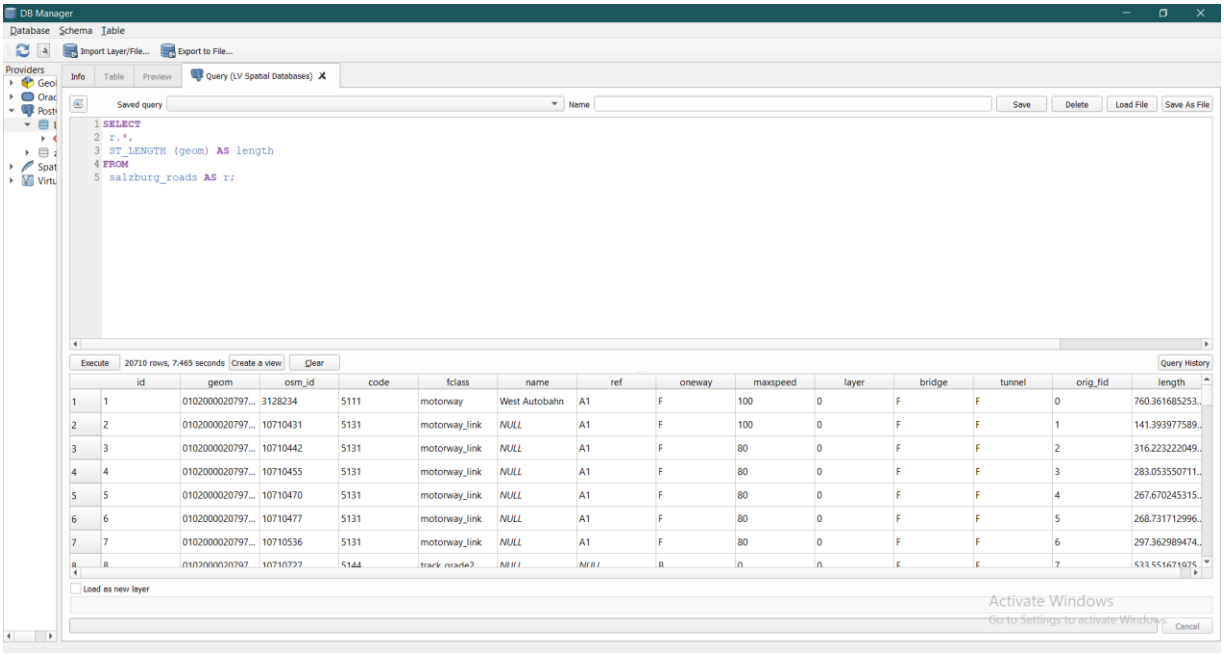
Kiarash Pooladsaz-12118707

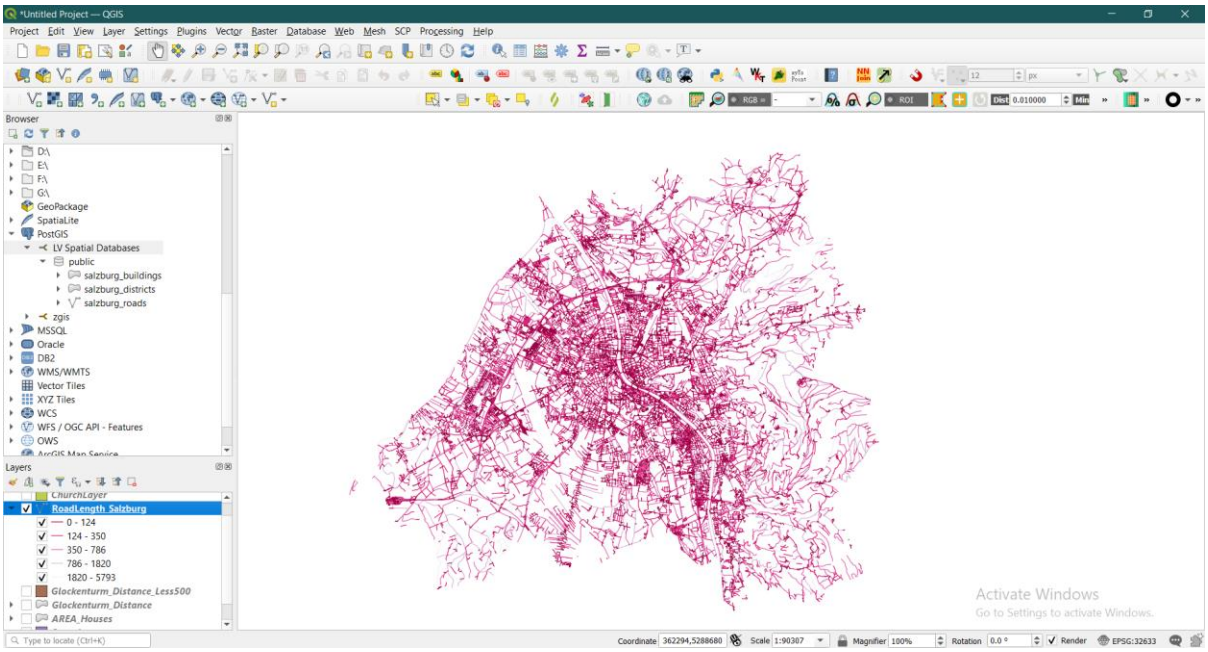
Spatial DataBases-SQL_III Assignment

- *Assign categories based on spatial properties*

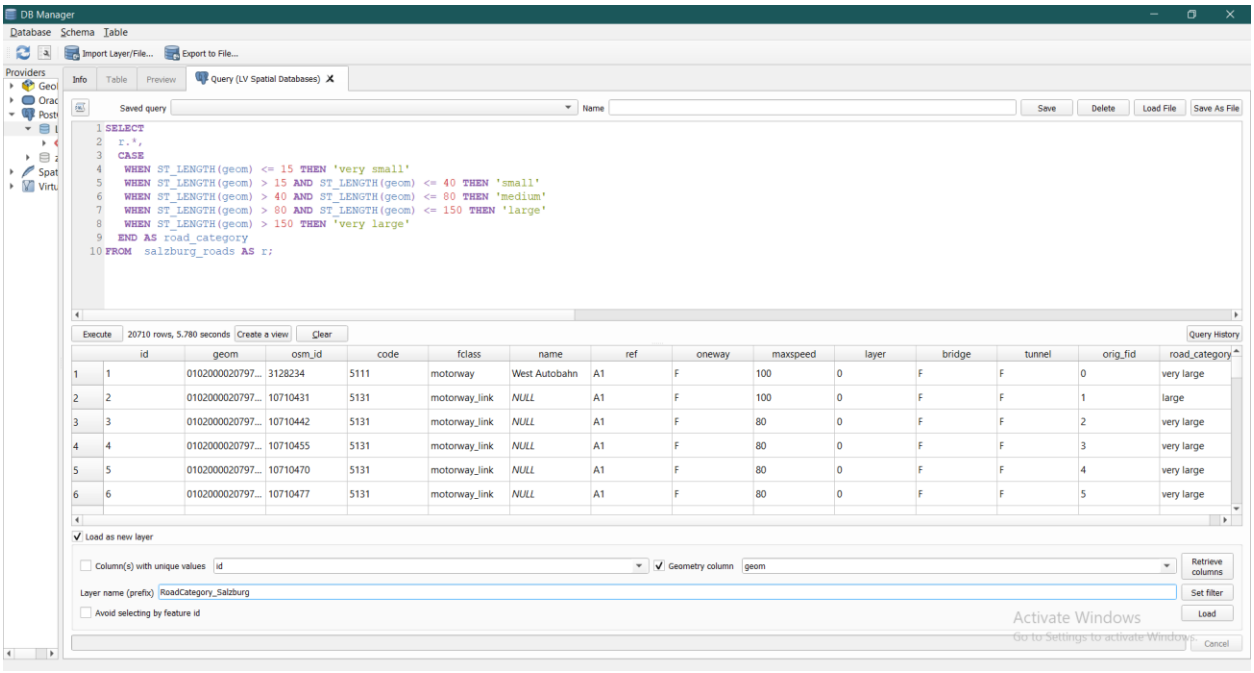
“Use SQL to create categories and associate numeric values of table columns to them, particularly categorizing roads based on their length.”

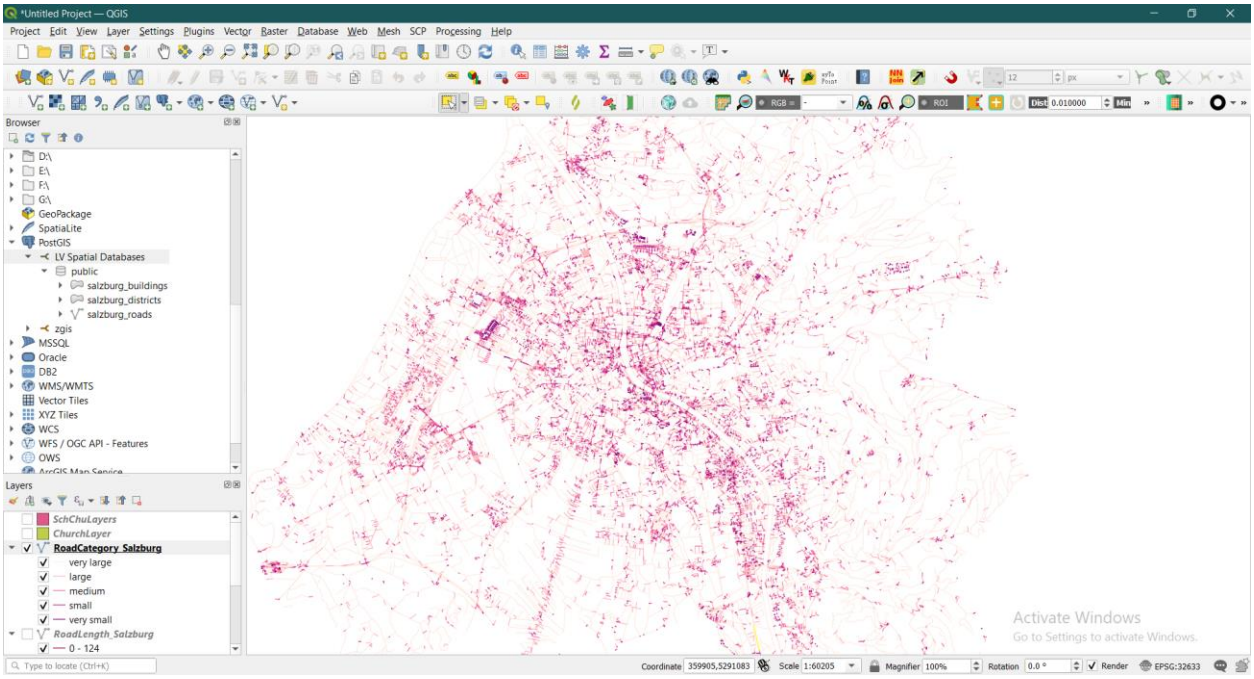
We can categorize the road layer length in 2 ways; we can calculate the length of each attribute and do symbology in Qgis, as follow:





Or, we can determine which amounts of length are being described as which type, such as small:

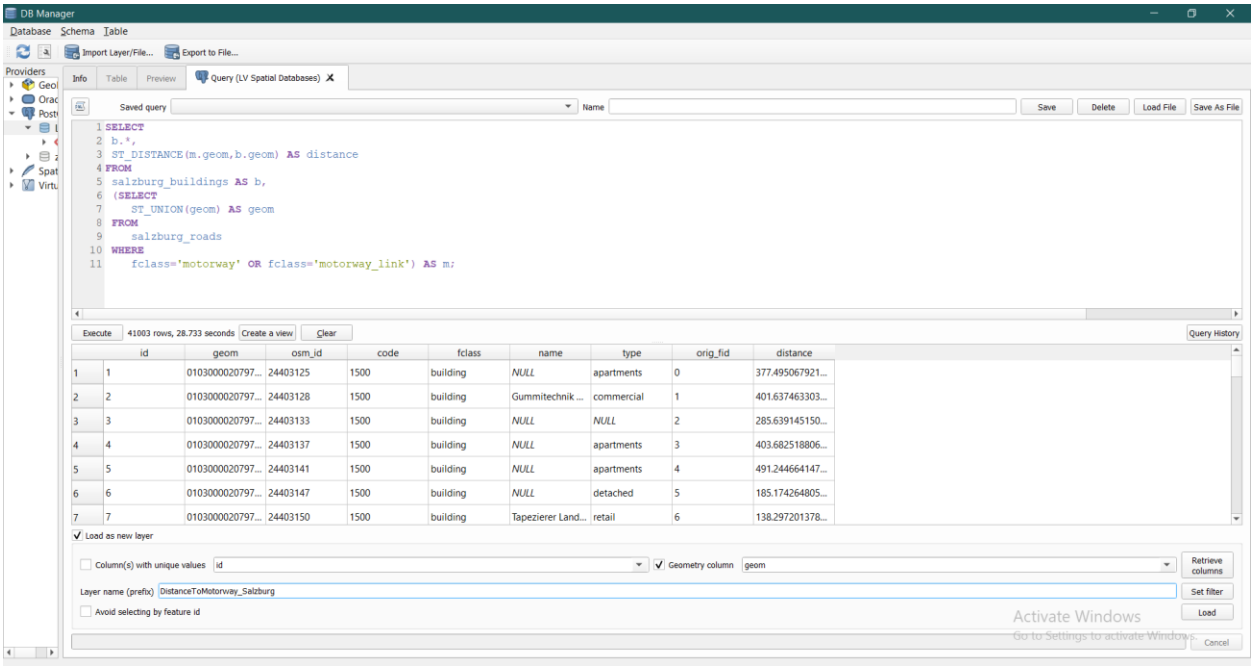




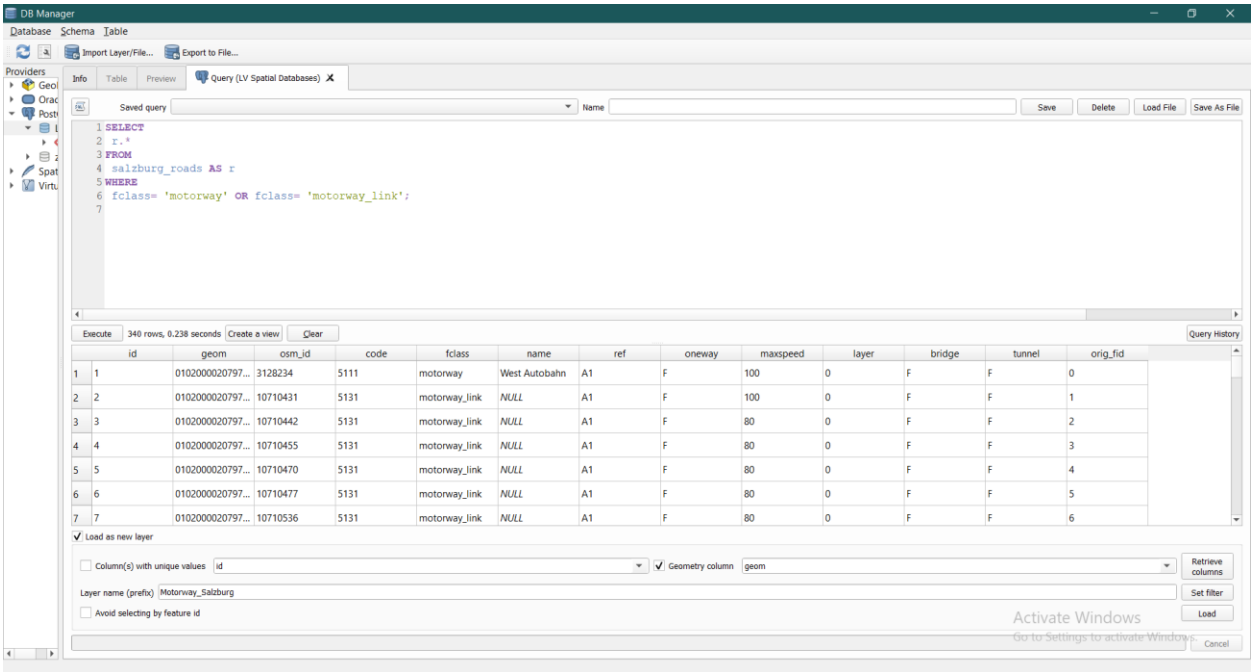
- *Cross-products with distance calculation*

“Create a query to calculate the distance from every building to a motorway.”

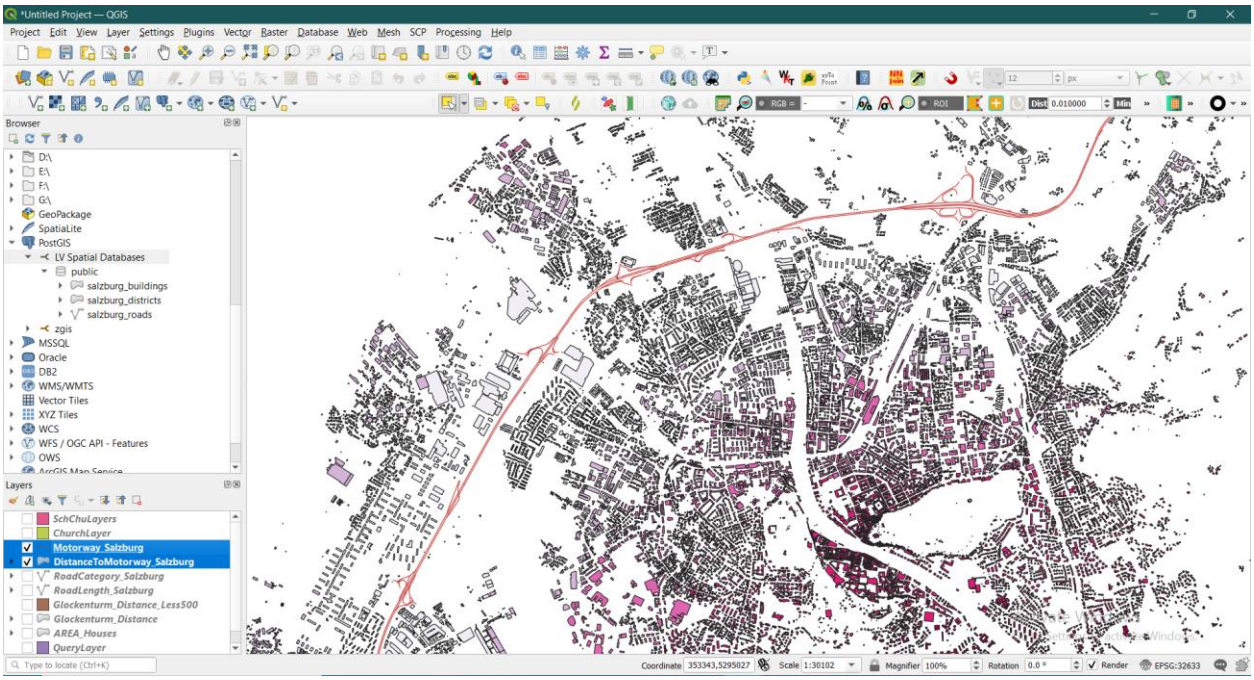
We should determine that we want to calculate the distance of one building to the closest motorway. Therefore, ST_UNION helps us here to avoid long-time waiting for other queries possibilities. It becomes the main measurement factor in this query instead of the default geom.



We also make a query to add motorway layer to the Qgis:



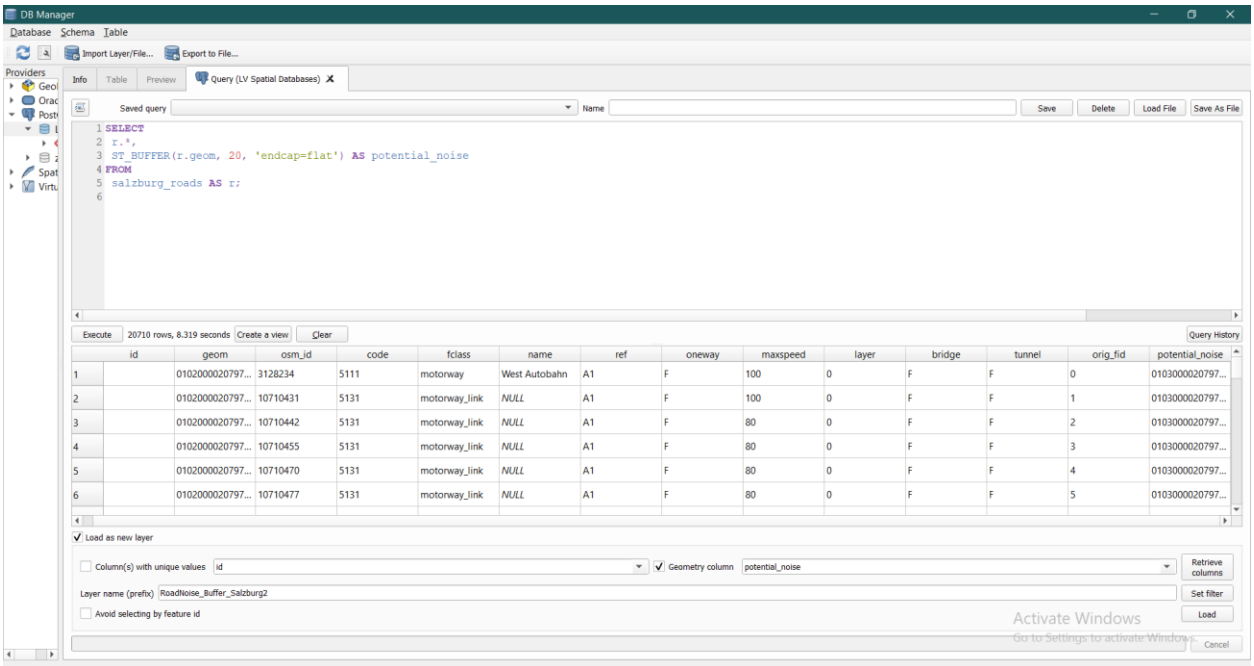
Therefore, the map shows what we were expecting☺:

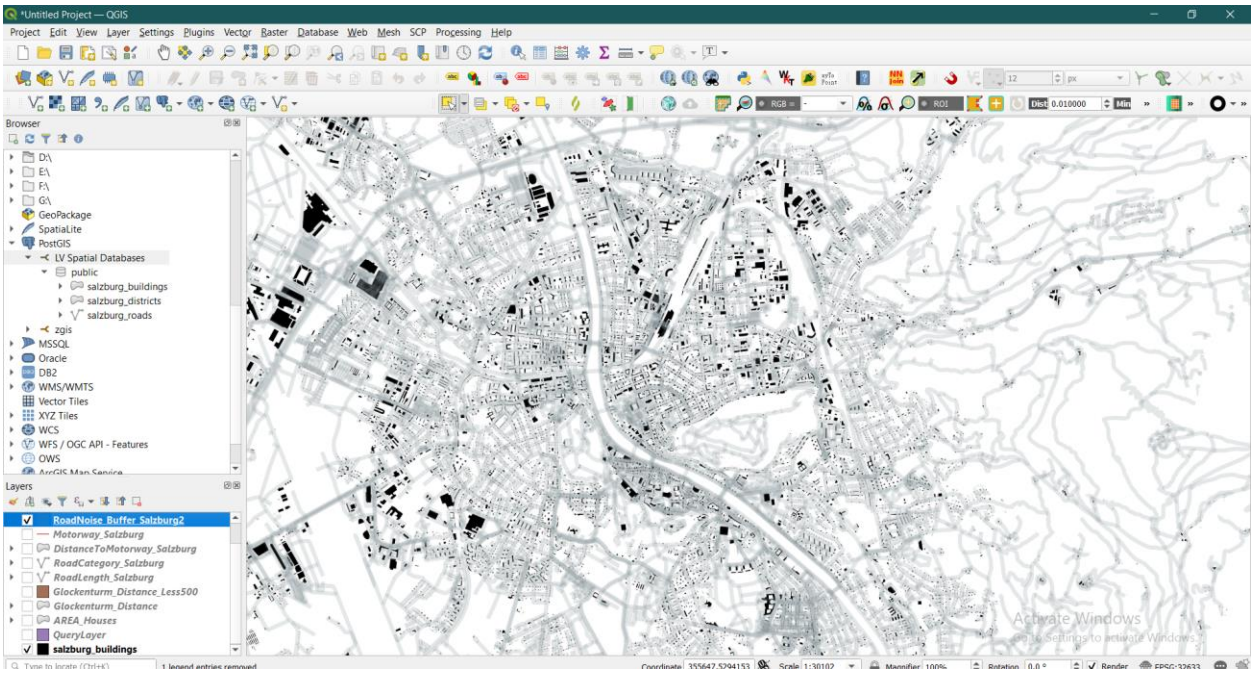


- *The particularities of buffers, negation, and long queries*

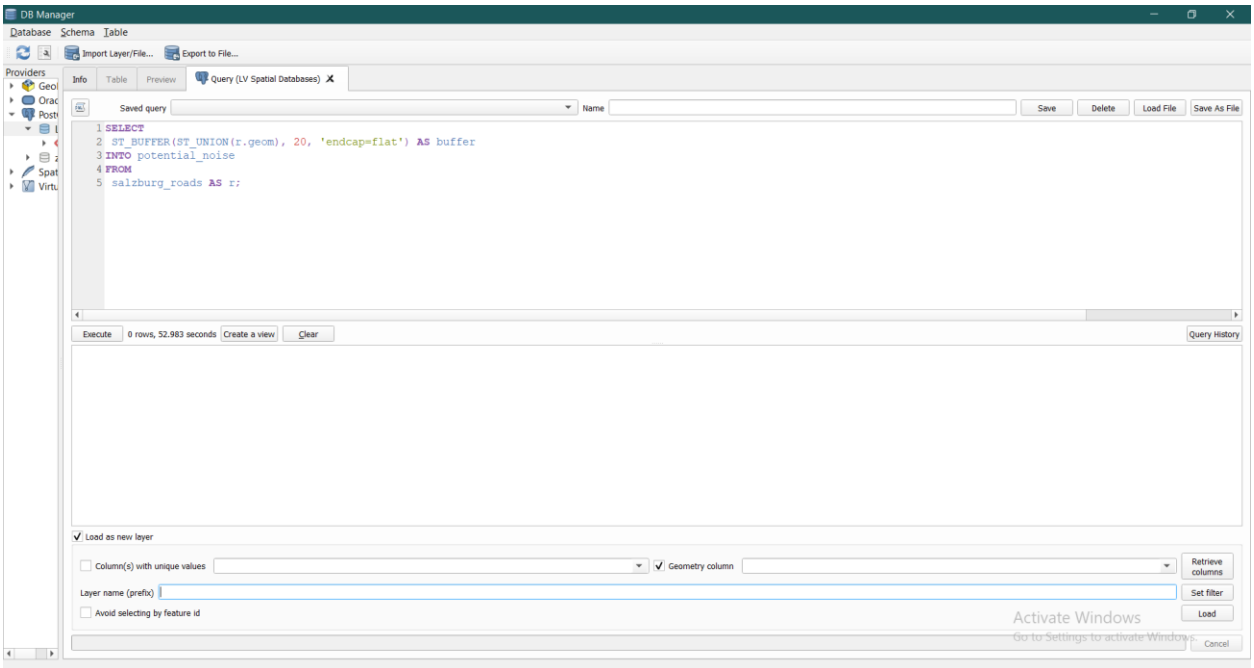
“Create a query that selects all buildings with more than 20 meters distance to a road.”

As mentioned in the assignment, there are 2 ways (maybe at least) to select buildings which are located more than 20 meters from the road. You can make a query by ST_BUFFER to create the potential noise buffer and then try to select buildings that their minimum distance to the 20 meter potential noise buffer is not true. Well, the buffer has been created first:

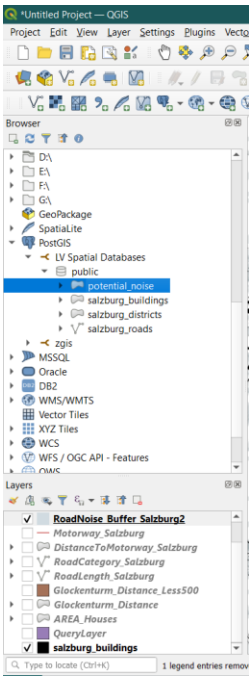




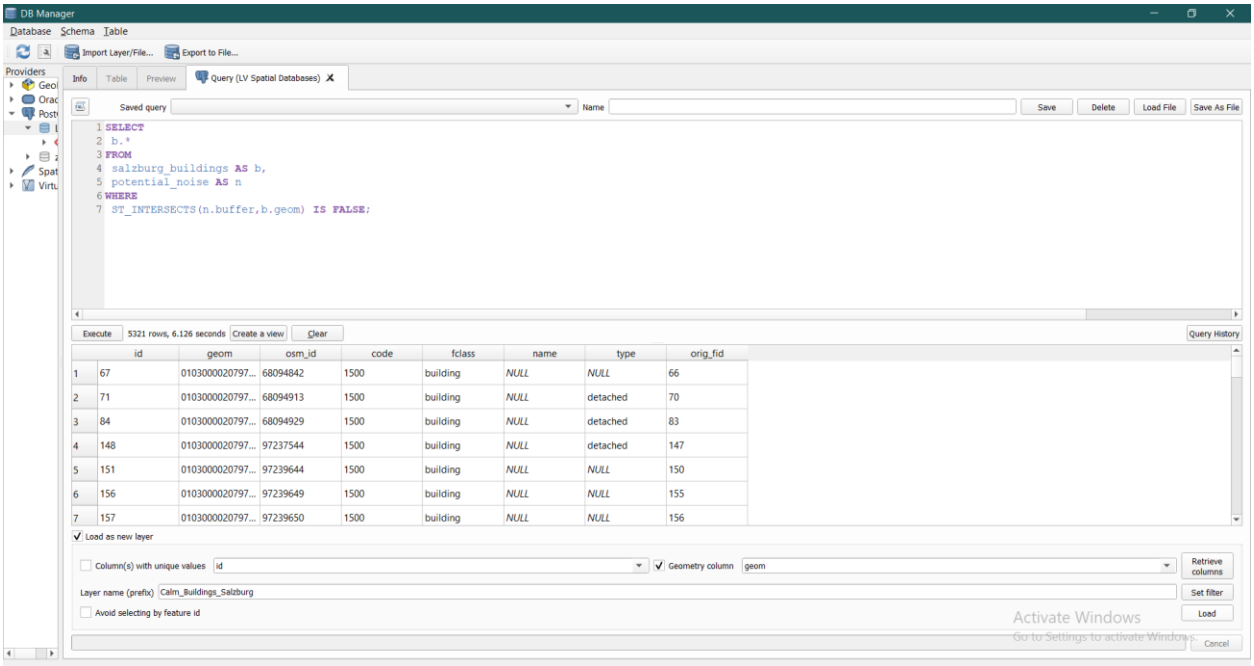
However, it seems that the performance of query would be so long for the next step! So, we can first create a table for the 20 meter potential noise buffer by SELECT…. INTO…. FROM…..

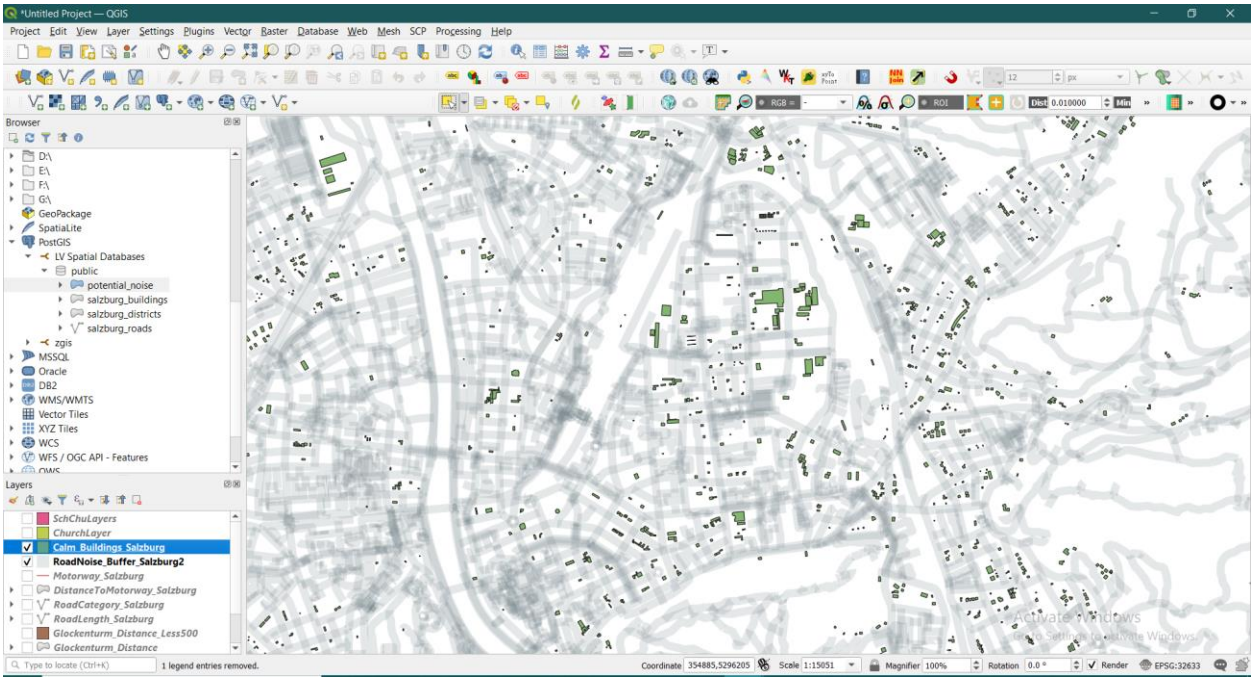


The created table is in the database now:



Then we intersect 2 tables, where the buildings with intersections with the buffer would be gone:





- *Think of three situations in which having GIS operations written as SQL queries can be beneficial and has advantages*

So, I can say if we imagine we have different type of shapefiles like point, polyline and polygon. Each could be like population and average income, road type, region. If you have a specific purpose, like the number of people in a certain region, living in 100-meter distance from the road, which consider all these 3 shapefiles together, SQL helps to decrease the computations for each shapefile and minimize the number of steps you need to pass.

Also, you can search easier for multiplied features in the layer with SQL, making distinct selection or skip them in some calculations. There may be some errors in the layer like some multiplied buildings attribute and you want to calculate the ratio of houses in each district. It certainly effect on the final value.