April 2024

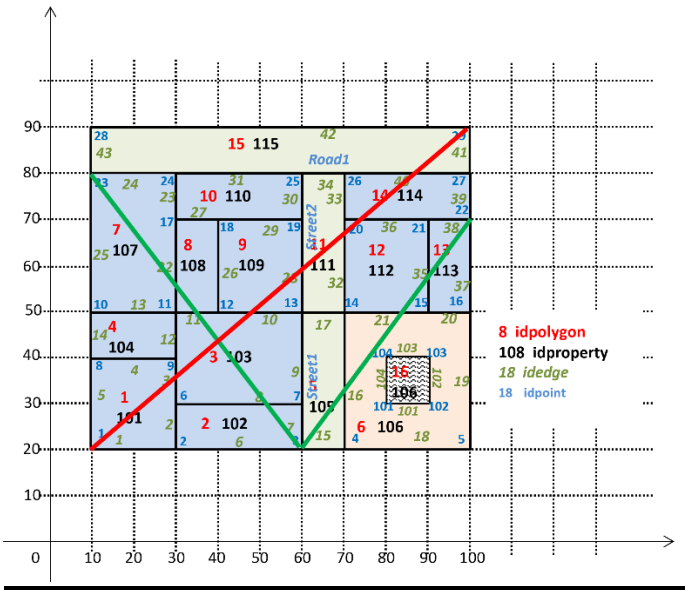
Exercise 2 Handling geometric elements with PostGIS

(submit – to the course site – short report with the code and the results until **13/5/2024)**

I. On the basis of the Elementary Cadaster of Exercise 1b, create tables point_g and polygon_g (corresponding to point and polygon) with the appropriate geometries. Then, enter the data of the point and polygon tables, respectively (the latter can be deleted). Test the functionality of the schema by formulating simple, spatial queries (similar to those in the examples presented in class.

Hint: For the point_g table use the create_point_g.sql document (located in the zipped folder with this documnet). Accordingly, for polygon_g, use the create_polygon_g.sql document. The sql queries of the two documents should be modified appropriately to work in schema **exercise1**. There are two ways to update the polygon_g table: (a) By manually entering the necessary UPDATE command for each polygon. (b) Automatically, with the special code [functions create_polygon_geom() and create_polygon_string()] of the second document. For the automatic mode to work, the entries in the polygonedge array must be entered for each polygon in the order of its perimeter continuity (the direction – clockwise or counter-clockwise – does not matter).

II. A water supply line is to be installed diagonally across the line (10 20, 100 90), as shown in Figure 2.1 below (red line). Create a table of line elements (line_g) and insert record for that line. Do the same for a second line (green), which passes through the points (10, 80), (60, 20) and (100,70).



Σχήμα 2.1

III. Formulate and execute appropriate queries for extracting and presenting the following data:

III.1 List of owners who are not your neighbors with the area of their property (if you do not have your own property, use polygon 10.

III.2 List of all pairs of adjacent polygons and the length of their common border, in order: (polygon1, polygon2, border), in ascending order of the identifiers of the first, in descending order of the border lengths and showing each pair only once.
Hint:
the combination of functions
ST_Length(ST_CollectionExtract(ST_Intersection(geometry1, geometry2), 2)) selects the line elements of the intersection (or contact) of two geometries and returns their length.

III.3 List of all polygons through which none of the two lines passes with the corresponding distances from them, in descending order of the distances from the red line.
Hint: the ST_DISTANCE(geometry1, geometry2) function returns the distance between two geometry elements.

III.4 Find the meeting point of the two water supply lines and the owners of the polygon that contains it.

III.5 List of owners, with the number of plots and the total area for each owner.