Zaawansowane technologie bazodanowe
Mikołaj Wielgos
Lab2

1. Utworzenie tabeli, pomiar odległości, układy odniesienia

```sql
CREATE TABLE cities
(
   id    serial,
   name  varchar(64),
   cords geometry
);

-- Warsaw
-- Coordinates: 52°13’48”N 21°00’40”E

-- Kraków
-- Coordinates: 50°03’41”N 19°56’14”E

INSERT INTO cities (name, cords)
VALUES ('Kraków', ST_GeomFromText('POINT(19.938333 50.061389)')),
      ('Warszawa', ST_GeomFromEWKT('SRID=4326;POINT(21.008333 52.232222)'));
SELECT *
FROM cities;

SELECT *, ST_SRID(cords)
FROM cities;

SELECT ST_AsEWKT(ST_Transform(cords, 2178))
FROM cities
WHERE id = 1;

SELECT ST_AsEWKT(ST_Transform(cords, 2178))
FROM cities
WHERE id = 2;

UPDATE cities
SET cords=ST_SetSRID(cords, 4326)
WHERE id = 1;
SELECT ST_AsEWKT(ST_Transform(cords, 2178))
FROM cities
WHERE id = 1;
```

```
UPDATE cities
SET cords_geog = cords::geography;
```

Wynik odległości w ćw. 2.

2. GEOMETRY vs. GEOGRAPHY

```
SELECT ST_DISTANCE(
            st_transform((SELECT cords FROM cities WHERE id = 1), 2178),
            st_transform((SELECT cords FROM cities WHERE id = 2), 2178))
        as distance_meters_2178,
    ST_DISTANCE(
            st_transform((SELECT cords FROM cities WHERE id = 1), 2180),
            st_transform((SELECT cords FROM cities WHERE id = 2), 2180))
        as distance_meters_2180,
    ST_DistanceSphere(
            (SELECT cords FROM cities WHERE id = 1),
            (SELECT cords FROM cities WHERE id = 2))
        as distance_meters_from_sphere,
    ST_DISTANCE(
            (SELECT cords_geog FROM cities WHERE id = 1),
            (SELECT cords_geog FROM cities WHERE id = 2), true)
        as distance_meters_geography_spheorid,
    ST_DISTANCE(
            (SELECT cords_geog FROM cities WHERE id = 1),
            (SELECT cords_geog FROM cities WHERE id = 2), false)
        as distance_meters_geography_spheroid;
```

```
+---------------------------------------+------------------+
|distance_meters_2178                   |252826.59901119964|
+---------------------------------------+------------------+
|distance_meters_2180                   |252696.8028960858 |
+---------------------------------------+------------------+
|distance_meters_from_sphere            |252654.4051553    |
+---------------------------------------+------------------+
|distance_meters_geography_spheorid     |252840.13360491   |
+---------------------------------------+------------------+
|distance_meters_geography_spheroid_false|252654.4051553    |
+---------------------------------------+------------------+
```

3. Analiza OpenStreetMap

```
function has_relevant_tags(tags)
   if tags.amenity == 'university' then return true end
   if tags.ref and tags.ref:find('C-2') then return true end
   if tags.ref and tags.ref:find('AGH') then return true end
   if tags.name and tags.name:find('Miasteczko Studenckie') then return true end
   if tags.short_name and tags.short_name:find('AGH') then return true end
   if tags.amenity == 'pub' then return true end
   if tags.amenity == 'bar' then return true end
   if tags.amenity == 'cafe' then return true end
   if tags.amenity == 'restaurant' then return true end
   if tags.amenity == 'toilets' then return true end
   return false
end
```

4. Import danych OSM

> osm2pgsql -H lab.kis.agh.edu.pl -d wlgs -U wlgs -P 1600 -W --schema osm1 -S ./flex.lua -O flex malopolskie-latest.osm.pbf
Password:
2024-11-08 07:33:41 osm2pgsql version 2.0.0
2024-11-08 07:33:41 Database version: 16.0 (Debian 16.0-1.pgdg110+1)
2024-11-08 07:33:41 PostGIS version: 3.4
2024-11-08 07:33:41 Initializing properties table '"osm1"."osm2pgsql_properties"'.
2024-11-08 07:33:41 Storing properties to table '"osm1"."osm2pgsql_properties"'.
2024-11-08 07:33:46 Reading input files done in 4s.
2024-11-08 07:33:46   Processed 20758351 nodes in 1s - 20758k/s
2024-11-08 07:33:46   Processed 2745972 ways in 2s - 1373k/s
2024-11-08 07:33:46   Processed 22491 relations in 1s - 22k/s
2024-11-08 07:33:46 No marked nodes or ways (Skipping stage 2).
2024-11-08 07:33:46 Clustering table 'points' by geometry...
2024-11-08 07:33:46 Clustering table 'lines' by geometry...
2024-11-08 07:33:46 Clustering table 'polygons' by geometry...
2024-11-08 07:33:46 Creating index on table 'points' ("geom")...
2024-11-08 07:33:46 Creating index on table 'lines' ("geom")...
2024-11-08 07:33:46 Creating index on table 'polygons' ("geom")...
2024-11-08 07:33:46 Analyzing table 'points'...
2024-11-08 07:33:46 Analyzing table 'lines'...
2024-11-08 07:33:46 Analyzing table 'polygons'...
2024-11-08 07:33:46 All postprocessing on table 'points' done in 0s.
2024-11-08 07:33:46 All postprocessing on table 'polygons' done in 0s.
2024-11-08 07:33:46 All postprocessing on table 'lines' done in 0s.
2024-11-08 07:33:46 Storing properties to table '"osm1"."osm2pgsql_properties"'.
2024-11-08 07:33:46 osm2pgsql took 5s overall.

5. Pole powierzchni AGH i MS

```
with areas as (SELECT name,
                      amenity,
                      ST_GeometryType(geom)                        as geom_type,
                      ST_Area(ST_Transform(geom, 2180)) / 1000000 as area_km2
               FROM osm1.polygons
               WHERE name LIKE '%AGH%'
                  OR name LIKE '%Miasteczko Studenckie%'
                  OR name LIKE '%Górniczo-%'
               ORDER BY name)
SELECT SUM(area_km2) as kampus_agh_area_km2
from areas;
```

```
+-------------------+
|kampus_agh_area_km2|
+-------------------+
|0.3422302365265838 |
+-------------------+
```

```
with areas as (SELECT name,
                      amenity,
                      ST_GeometryType(geom)                        as geom_type,
                      ST_Area(ST_Transform(geom, 2180)) / 1000000 as area_km2
               FROM osm1.polygons
               WHERE name LIKE '%Miasteczko Studenckie%'
               ORDER BY name)
SELECT SUM(area_km2) as ms_agh_area_km2
from areas;
```

```
+-------------------+
|ms_agh_area_km2    |
+-------------------+
|0.09199785802817415|
+-------------------+
```

6. Puby i toalety na AGH

```
WITH areas AS (SELECT name,
                      geom
               FROM osm1.polygons
               WHERE name LIKE '%AGH%'
                  OR name LIKE '%Miasteczko Studenckie%'
                  OR name LIKE '%Górniczo-%'),
    points_of_interest AS (SELECT name as object_name,
                                  amenity,
                                  geom
                           FROM osm1.points
                           WHERE amenity IN ('pub', 'toilets')
                           UNION ALL
                           SELECT name              as object_name,
                                  amenity,
                                  ST_Centroid(geom) as geom
                           FROM osm1.polygons
                           WHERE amenity IN ('pub', 'toilets'))
SELECT a.name as area_name,
      p.amenity as object_type,
      COALESCE(p.object_name, 'unnamed') as object_name
FROM areas a
       JOIN points_of_interest p
            ON ST_Contains(a.geom, p.geom)
ORDER BY a.name,
       p.amenity,
       p.object_name;
```

```
+-------------------------+-----------+--------------+
|area_name                |object_type|object_name   |
+-------------------------+-----------+--------------+
|Akademia Górniczo-Hutnicza|pub        |Gwarek        |
|Akademia Górniczo-Hutnicza|pub        |Karlik        |
|Miasteczko Studenckie AGH |pub        |Klub Filutek  |
|Miasteczko Studenckie AGH |pub        |Klub Zaścianek|
|Miasteczko Studenckie AGH |toilets    |unnamed       |
+-------------------------+-----------+--------------+
```

7. Dokąd na piwo?

```
WITH building_c2 AS (SELECT geom
                     FROM osm1.polygons
                     WHERE ref = 'C-2')
SELECT p.node_id,
      p.name                                                as pub_name,
      ST_Distance(
              ST_Transform(p.geom, 2180),
              ST_Transform(b.geom, 2180)
      )                                                     as distance,
      ST_AsGeoJSON(ST_Transform(p.geom, 4326))::json as geom
FROM osm1.points p
        CROSS JOIN building_c2 b
WHERE p.amenity = 'pub'
 AND p.name IS NOT NULL
 AND ST_DWithin(
       ST_Transform(p.geom, 2180),
       ST_Transform(b.geom, 2180),
       1000
     )
ORDER BY distance;
```

```
+----------+----------------------------+----------------+-----------------------------------------------------+
|osm_id    |pub_name                    |distance        |geom                                                 |
+----------+----------------------------+----------------+-----------------------------------------------------+
|2351408882|Gwarek                      |439.8954416528079|{"type":"Point","coordinates":[19.915753,50.065825999]} |
|6148239779|Whisky Bar Egon             |487.9391320968242|{"type":"Point","coordinates":[19.9183065,50.069828699]}|
|8757494572|Cosmic Games Pub            |505.2485320327163|{"type":"Point","coordinates":[19.9301169,50.066220799]}|
|2351408470|Karlik                      |529.2612405839083|{"type":"Point","coordinates":[19.9145004,50.066085999]}|
|9894718217|Świat Piwa. Beer Shop & Bistro|561.15602034238  |{"type":"Point","coordinates":[19.9307407,50.064829199]}|
|752177457 |Garage Pub                  |706.2266801993362|{"type":"Point","coordinates":[19.9263137,50.059768299]}|
|7049970185|Klub Buda                   |743.021848267166 |{"type":"Point","coordinates":[19.930108,50.060879299]} |
|3288623107|Zaginiony Świat             |765.0430554136412|{"type":"Point","coordinates":[19.933319,50.067862499]} |
|279040195 |Stary Port                  |804.4231639763722|{"type":"Point","coordinates":[19.9318205,50.061259399]}|
|2907528026|The Stage                   |812.2527690792697|{"type":"Point","coordinates":[19.9343907,50.066468799]}|
+----------+----------------------------+----------------+-----------------------------------------------------+
```