

Zadanie 3

Autor: Samuel Schmidt

AIS id: 103120

Uloha 2

Zadanie: zistite aké kraje sú na Slovensku (planet_osm_polygon, admin_level = '4') a vypíšte ich súradnice ako text s longitude a latitude.

	name text	coordinates text	way geometry
1	Prešovský kraj	19.883928 49.20417799964193,19.883958 49.2041779996419...	0103000020110F0000010000004550000AE07B65E2EE34041D3A9295AB2115841CF8A2D0A3...
2	Košický kraj	20.18122199999998 48.74099699975657,20.181416 48.7405...	0103000020110F00000100000098470000CDEBA5ADD12341412E3E96EEFCC457419408EF79D...
3	Banskobystrický kraj	18.47867299999997 48.54860199980419,18.478683 48.5485...	0103000020110F000001000000644D00009F8709784C633F41F68D0B3455A557410CDF03954...
4	Trenčiansky kraj	17.35303569999996 48.7775265997475,17.353104 48.77746...	0103000020110F00000100000048390000F0230C19D3793D41916CAC1103CB5741F04A72B3D...
5	Bratislavský kraj	16.83318909999995 48.380524099845566,16.8332193 48.38...	0103000020110F0000010000008C1C0000C573FE09C6973C4198F047E5C68957414390A066C...
6	Trnavský kraj	16.933595 48.600625999791326,16.93359899999997 48.600...	0103000020110F0000010000006B4200003509362C6FC33C41D5CAC96BE1AD5741C6C5339E...
7	Žilinský kraj	18.322605 49.31510899961467,18.32267499999997 49.3150...	0103000020110F000001000000E23500005CD2000F6F1F3F4191CA03BB2C2458414D36D9D97...
8	Nitriansky kraj	17.705419 47.75898499999446,17.70550769999995 47.7588...	0103000020110F0000010000004370000D95D343A0E133E41EA72DFF1A82457412560F5191...

V selecte som pretransformoval dátá z geometry na geography pomocou funkcie ST_Transform. Následne som použil funkciu ST_AsText na transformáciu geography na čitelné súradnice long/lat. Substring len zastáva funkcionality pre krajší výpis.

```
SELECT name, substring(left(ST_ASTEXT(ST_Transform(way, 4326)::geography ), -2), 10)
as coordinates, way
FROM planet_osm_polygon WHERE admin_level='4';
```

Uloha 3

Zadanie: zoradťte kraje podľa ich veľkosti (st_area). Veľkosť vypočítajte pomocou vhodnej funkcie a zobrazte v km² v SRID 4326.

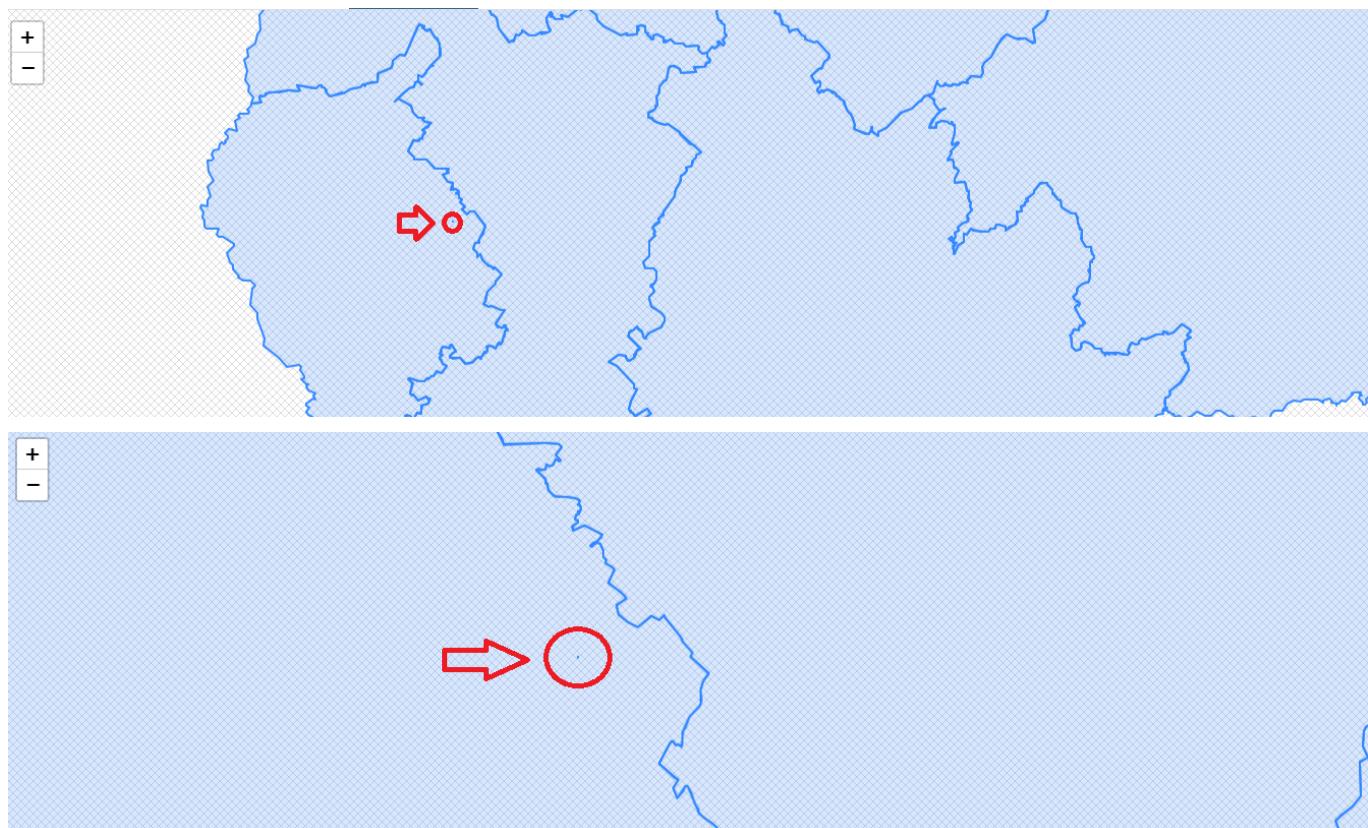
	name text	area double precision	way geometry
1	Bratislavský kraj	2051.660249334432	0103000020110F0...
2	Trnavský kraj	4145.350058274841	0103000020110F0...
3	Trenčiansky kraj	4501.795536203455	0103000020110F0...
4	Nitriansky kraj	6341.240667758643	0103000020110F0...
5	Košický kraj	6751.964233845743	0103000020110F0...
6	Žilinský kraj	6806.886696700907	0103000020110F0...
7	Prešovský kraj	8971.62395364223	0103000020110F0...
8	Banskobystrický kraj	9454.546380914244	0103000020110F0...

Pre výpočet plochy som použil funkciu ST_Area. Pre čo najpresnejší výsledok som previedol way(geometry) na geography pomocou funkcie ST_Transform. Kraje sa nachádzajú pod admin_level 4.

```
SELECT name, (ST_Area(ST_Transform(way, 4326)::geography))/10^6 as area, way
FROM planet_osm_polygon WHERE admin_level='4' ORDER BY area;
```

Uloha 4

Zadanie: pridajte si dom, kde bývate ako polygón (najdite si súradnice napr. cez google maps) do planet_osm_polygon (znova pozor na súradnicový systém). Výsledok zobrazte na mape.



Zvolil som si môj rodný dom v Budmericiach (nachádzajú sa na hranici Bratislavského kraja s Trnavským krajom). Zvolil som dva obrázky aj pre priblížený pohľad pre lepšiu vizualizáciu. Použil som pri vkladaní long/lat údaje geography, ktorú som previedol na geometry (typ stĺpca way).

```
Insert into planet_osm_polygon (name, way)
values ('house', ST_ASTEXT(ST_Transform(ST_GeogFromText('SRID=4326; POLYGON((17.403444019918297
48.35877744587877,
17.403519792324815
48.35872487002377,
17.403624398478946
48.35877566364728,
17.403541249997456
48.358829130564686,
17.403444019918297
48.35877744587877))')::geometry, 3857)));
```

```
SELECT name, substring(left(ST_ASTEXT(ST_Transform(St_GeomFromText(ST_AsText(way),  
3857), 4326)), -2), 10) as coordinates, way  
FROM planet_osm_polygon WHERE admin_level='4' or name = 'house';
```

Uloha 5

Zadanie: zistite v akom kraji je váš dom.

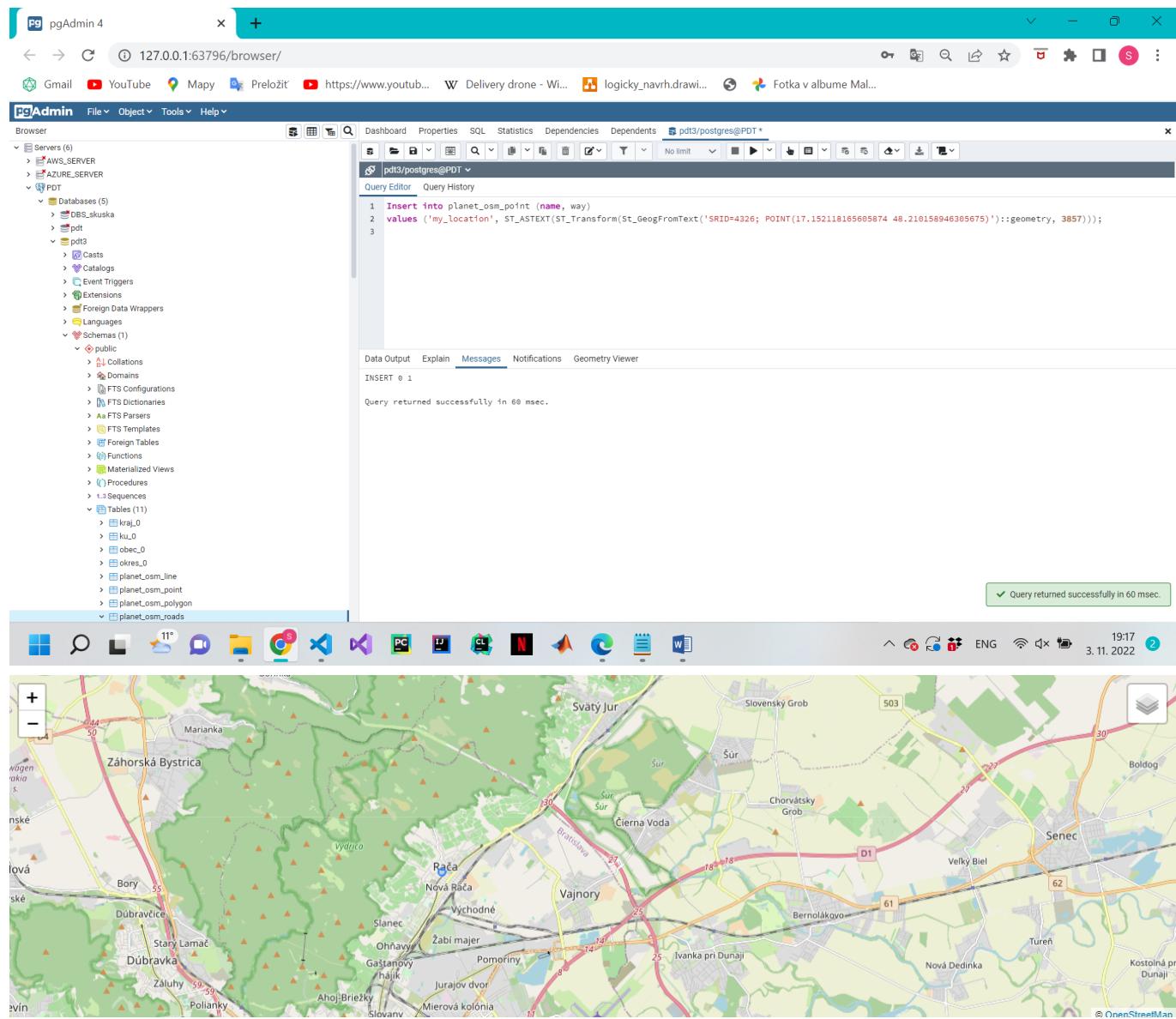
	name	text	lock
1	Bratislavský kraj		

Najskôr som vybral môj dom a joinol k nemu všetky kraje podľa ich spoločného prieniku. Nakoniec som z tohto joinu vytiahol meno kraja v ktorom sa môj dom nachádza.

```
SELECT name FROM planet_osm_polygon  
join (select ST_ASTEXT(way) as way FROM planet_osm_polygon WHERE name = 'house')  
as house on ST_intersects(house.way, ST_ASTEXT(planet_osm_polygon.way))  
where planet_osm_polygon.admin_level = '4';
```

Uloha 6

Zadanie: pridajte si do planet_osm_point vašu aktuálnu polohu (pozor na súradnicový systém). Výsledok zobrazte na mape.



Pre túto úlohu som zvolil polohu bytu v Rači. Údaje boli long/lat, čiže ich bolo potrebné opäť previesť do geometrie.

```

Insert into planet_osm_point (name, way)
values ('my_location', ST_ASTEXT(ST_Transform(St_GeogFromText('SRID=4326;
POINT(17.152118165605874 48.210158946305675'))::geometry, 3857)));

```

Uloha 7

Zadanie: zistite či ste doma - či je vaša poloha v rámci vášho bývania.

```
1 SELECT ST_TOUCHES((select way FROM planet_osm_point WHERE name = 'my_location'), (select way FROM planet_osm_polygon WHERE name = 'house'));
```

Data Output Explain Messages Notifications Geometry Viewer

st_touches	boolean
1	false

Zvolil som funkciu ST_Touches pre zistenie, či moja polohu (Rača) sa nachádza u mňa doma (Budmerice).

Nezvolil som funkciu WITHIN najmä preto, že táto funkcia neberie krajné body, čiže ak by som sa nachádzal na hranici môjho domu, výsledok by bol false (čo by bolo nesprávne). Výsledok bol teda false keďže Rača != Budmerice.

```
SELECT ST_TOUCHES((select way FROM planet_osm_point WHERE name = 'my_location'),
(select way FROM planet_osm_polygon WHERE name = 'house'));
```

Uloha 8

Zadanie: zistite ako ďaleko sa nachádzate od FIIT (name = 'Fakulta informatiky a informačných technológií STU'). Pozor na správny súradnicový systém – vzdialenosť musí byť skutočná.

```
1 SELECT ST_DISTANCE((select St_Transform(way, 4326)::geography FROM planet_osm_point WHERE name = 'my_location'),
2                     (select St_Transform(way, 4326)::geography FROM planet_osm_polygon WHERE name = 'Fakulta informatiky a informačných technológií STU'));
```

Data Output Explain Messages Notifications Geometry Viewer

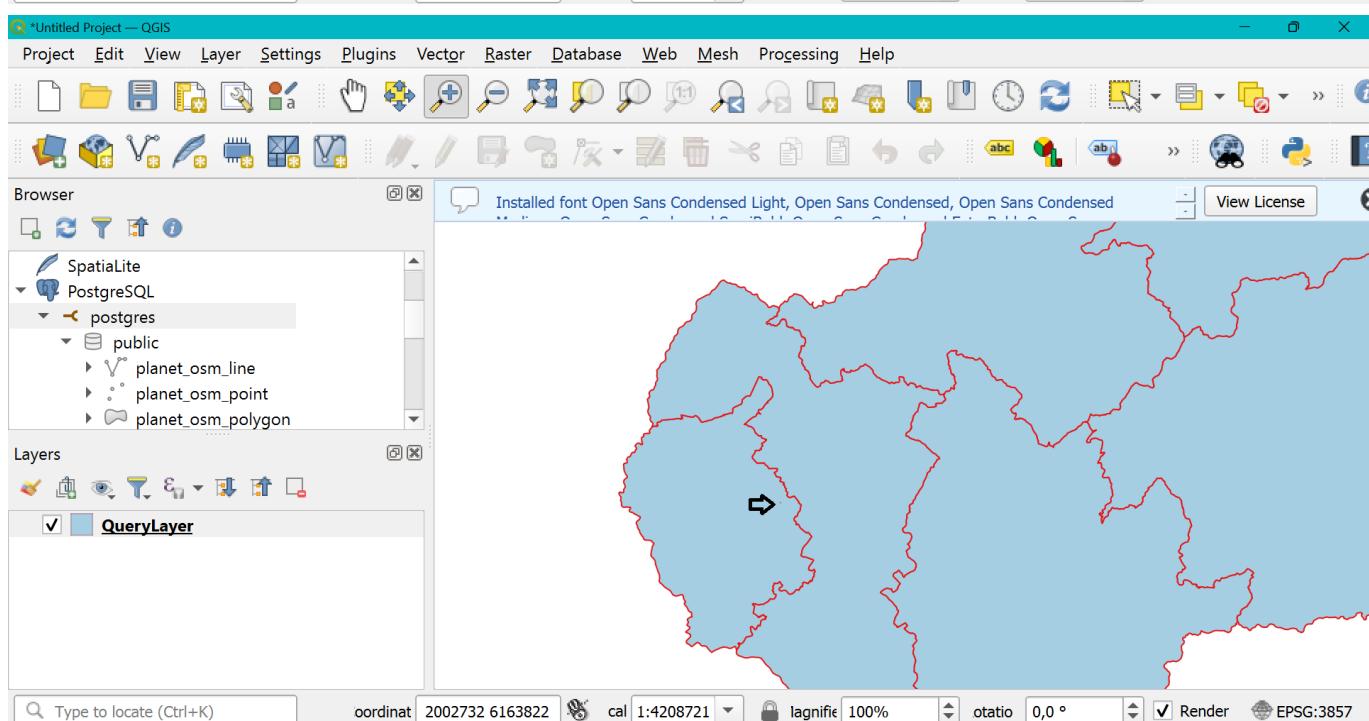
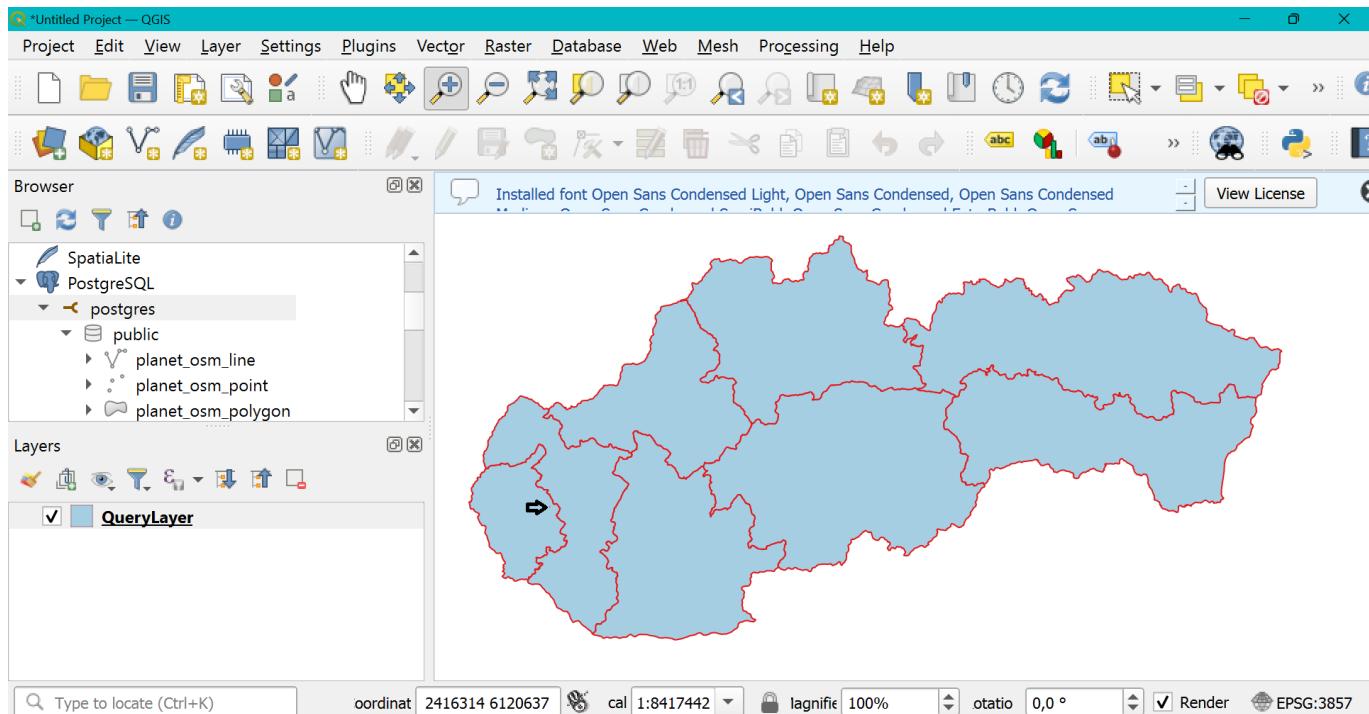
st_distance	double precision
1	8585.80627977

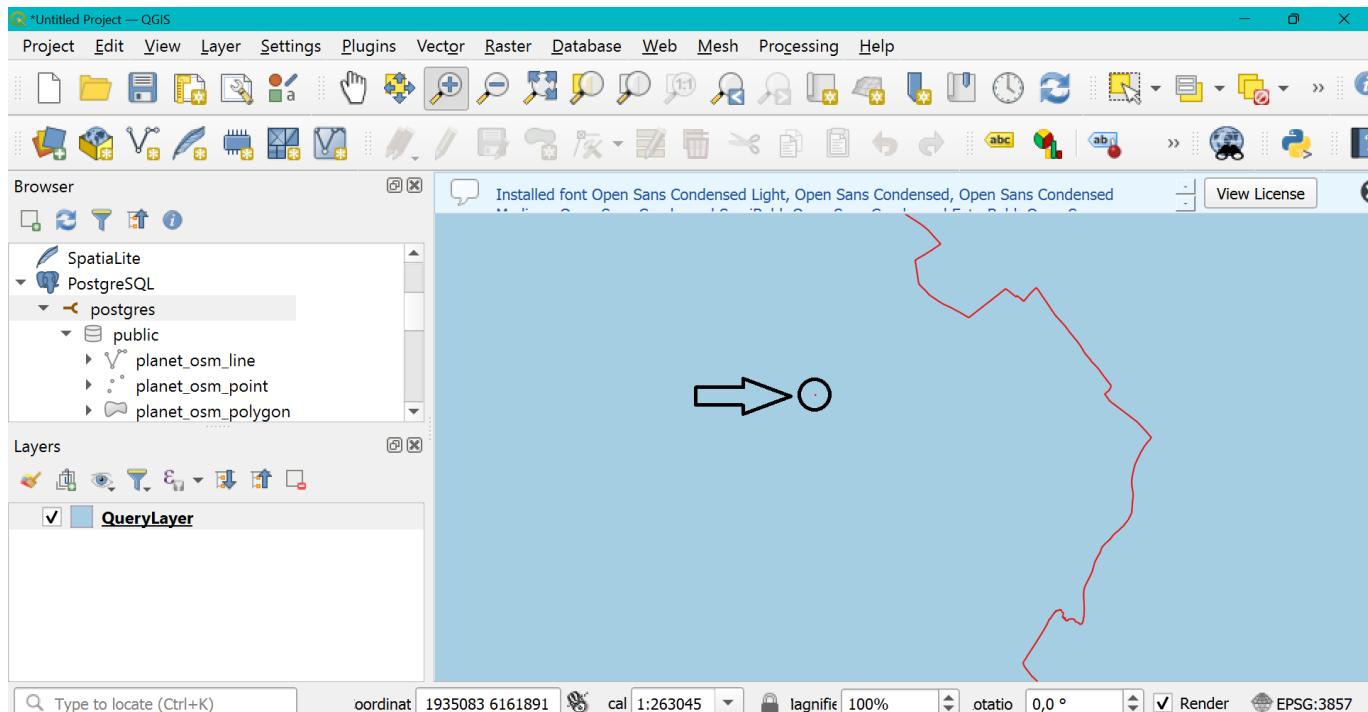
Pre správne riešenie úlohy som musel súradnice transformovať na geografiu, nakoľko geometria vracala nesprávny výsledok, keďže nezohľadňuje napríklad zakrivenie zeme.

```
SELECT ST_DISTANCE((select St_Transform(way, 4326)::geography FROM
planet_osm_point WHERE name = 'my_location'),
(select St_Transform(way, 4326)::geography FROM
planet_osm_polygon WHERE name = 'Fakulta informatiky a informačných technológií
STU'));
```

Uloha 9

Zadanie: Stiahnite si QGIS a vyplotujte kraje a váš dom z úlohy 2 na mape - napr. červenou čiarou.





Výsledok je rovnaký ako pre úlohu 2, vo vizualizácii som použil 3 obrázky pre lepšie priblíženie výsledku.

```
select name, way from planet_osm_polygon where name = 'house' or admin_level = '4'
```

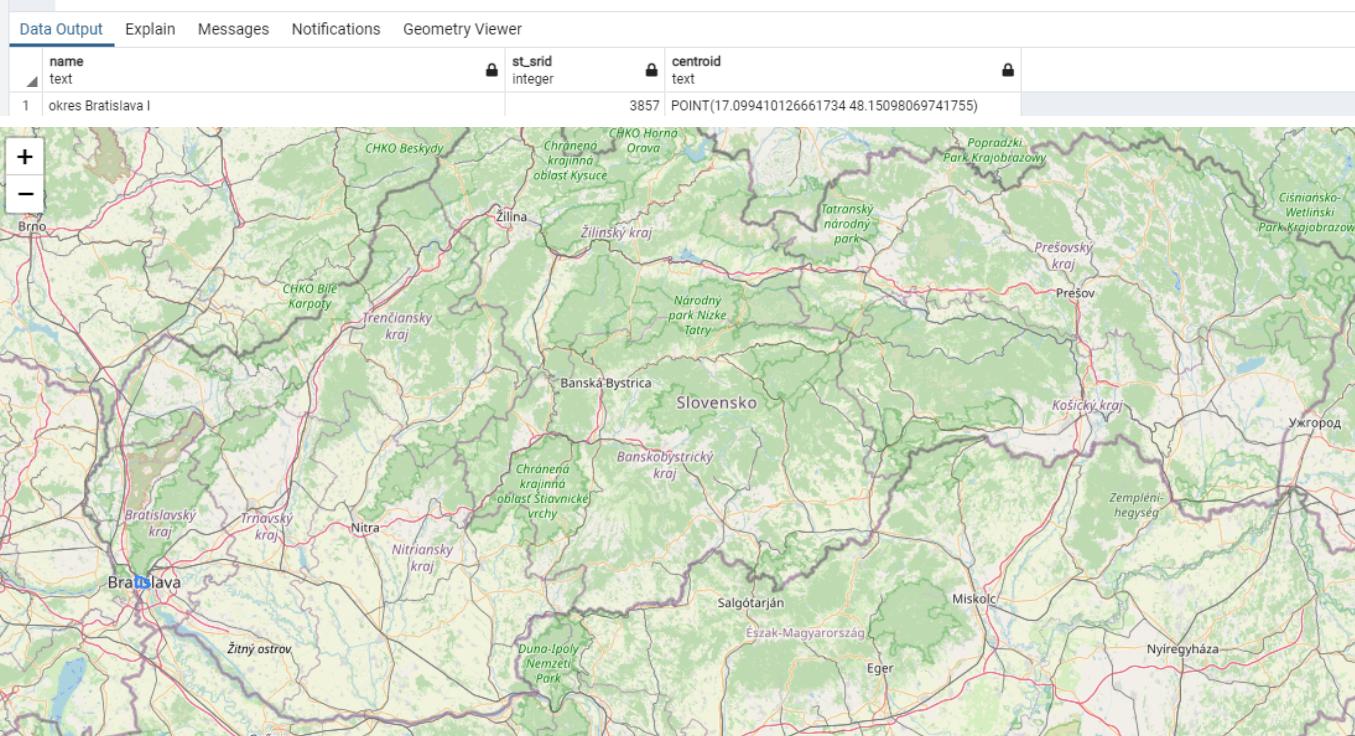
Uloha 10

Zadanie: Zistite súradnice centroidu (ťažiska) plošne najmenšieho okresu (vo výsledku nezabudnite uviesť aj EPSG kód súradnicového systému).

```

1 WITH region AS (SELECT name, way, (ST_Area(ST_Transform(way, 4326)::geography))/10^6 AS area
2   FROM planet_osm_polygon
3   WHERE admin_level='8' AND ST_WITHIN(way, (SELECT way FROM planet_osm_polygon WHERE admin_level='2' and name='Slovensko'))
4   ORDER BY area ASC
5   LIMIT 1)
6
7 SELECT region.name, ST_SRID(region.way),
8 ST_ASTEXT(
9   ST_transform(
10    ST_centroid((SELECT region.way FROM region)), 4326
11   )
12 ) AS centroid FROM region;

```



Najskôr som našiel najmenší okres pomocou príeniku so Slovenskom (pretože najmenší údaj pri okresoch = admin_level 8 bol kostol v maďarsku). Z neho som následne vyselektoval plochu (prevedenú do geografie). Z nej pomocou ST_Centroid vyselektujem centroid.

```

WITH region AS (SELECT name, way, (ST_Area(ST_Transform(way,
4326)::geography))/10^6 AS area
   FROM planet_osm_polygon
   WHERE admin_level='8' AND ST_WITHIN(way, (SELECT way FROM
planet_osm_polygon WHERE admin_level='2' and name='Slovensko'))
   ORDER BY area ASC
   LIMIT 1)

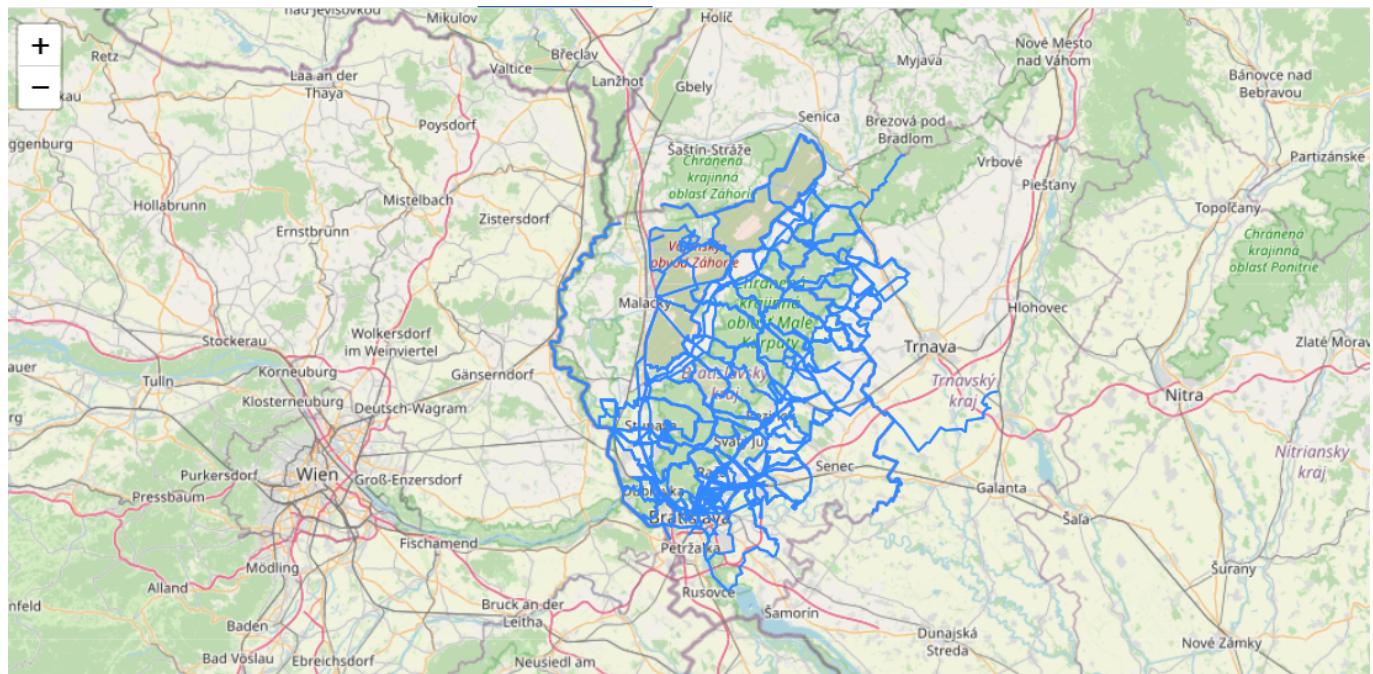
SELECT region.name, ST_SRID(region.way), ST_Transform(region.way,
4326)::geography,
ST_ASTEXT(
  ST_transform(
    ST_centroid((SELECT region.way FROM region)), 4326

```

```
)
) as centroid FROM region;
```

Uloha 11

Zadanie: Vytvorte priestorovú tabuľku všetkých úsekov cest, ktorých vzdialenosť od vzájomnej hranice okresov Malacky a Pezinok je menšia ako 10 km.



V tejto úlohe som si najskôr vytvoril tabuľku, ktorá obsahuje len 3 základné info, ktoré potrebujem a to osm_id = aby každý záznam mal id, name= meno cesty, way= geometrické údaje. Najskôr som si našiel hranicu medzi Malackami a Pezinkom pomocou Intersection (ich prienik bude len ciara). Potom som si nasiel všetky cesty vo vzdialosti 10km od tejto hranice. Použil som opäť geografiu pre presnejšie výsledky. Nebral som do úvahy flagy pre rôzne druhy cest a to z toho dôvodu, že databáza bola pomerne rozbitá a nedalo sa z nej vyselektovať všetky rozumné cesty jednoduchým flagom. Dá sa teda tato query upresniť podľa potreby pri výbere cest a to podmienkou WHERE za posledným joinom v sql kóde uvedenom nižšie (napr. WHERE highway is not null).

```

CREATE TABLE IF NOT EXISTS public.roads_MA_PE
(
    osm_id bigint,
    name text COLLATE pg_catalog."default",
    way geometry(LineString,3857)
);

ALTER TABLE public.planet_osm_roads
OWNER to postgres;

CREATE INDEX MA_PE_roads_way_idx
ON public.roads_MA_PE USING gist
(way)
WITH (FILLCOMPCT=100)
```

```

TABLESPACE pg_default;

INSERT INTO roads_MA_PE(
WITH border as(
SELECT ST_TRANSFORM(ST_INTERSECTION(
(SELECT way FROM planet_osm_polygon WHERE admin_level = '8' and name = 'okres Malacky'),
(SELECT way FROM planet_osm_polygon WHERE admin_level = '8' and name = 'okres Pezinok')
),4326)::geography as transformed
)
SELECT osm_id, name, way FROM planet_osm_roads as roads
join border on ST_DISTANCE(border.transformed, ST_TRANSFORM(roads.way,
4326)::geography) < 10000);

```

Uloha 12

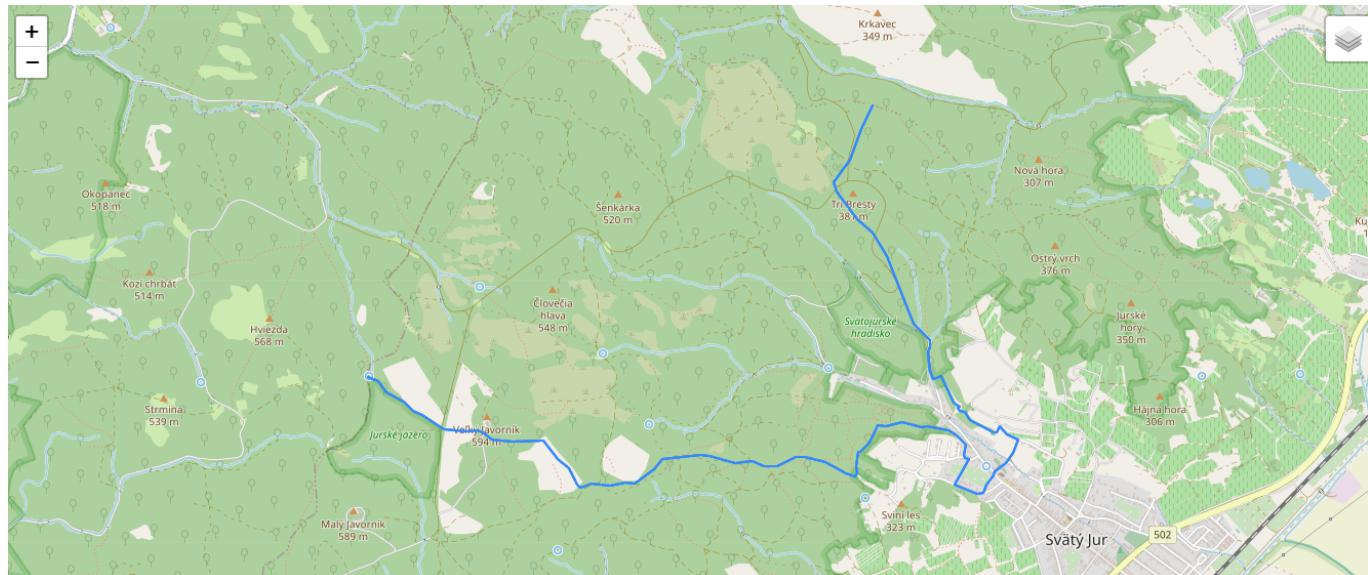
Zadanie: Jedným dopytom zistite číslo a názov katastrálneho územia (z dát ZBGIS, https://www.geoportal.sk/sk/zbgis_smd/na-stiahnutie/), v ktorom sa nachádza najdlhší úsek cesty (z dát OSM) v okrese, v ktorom bývate.

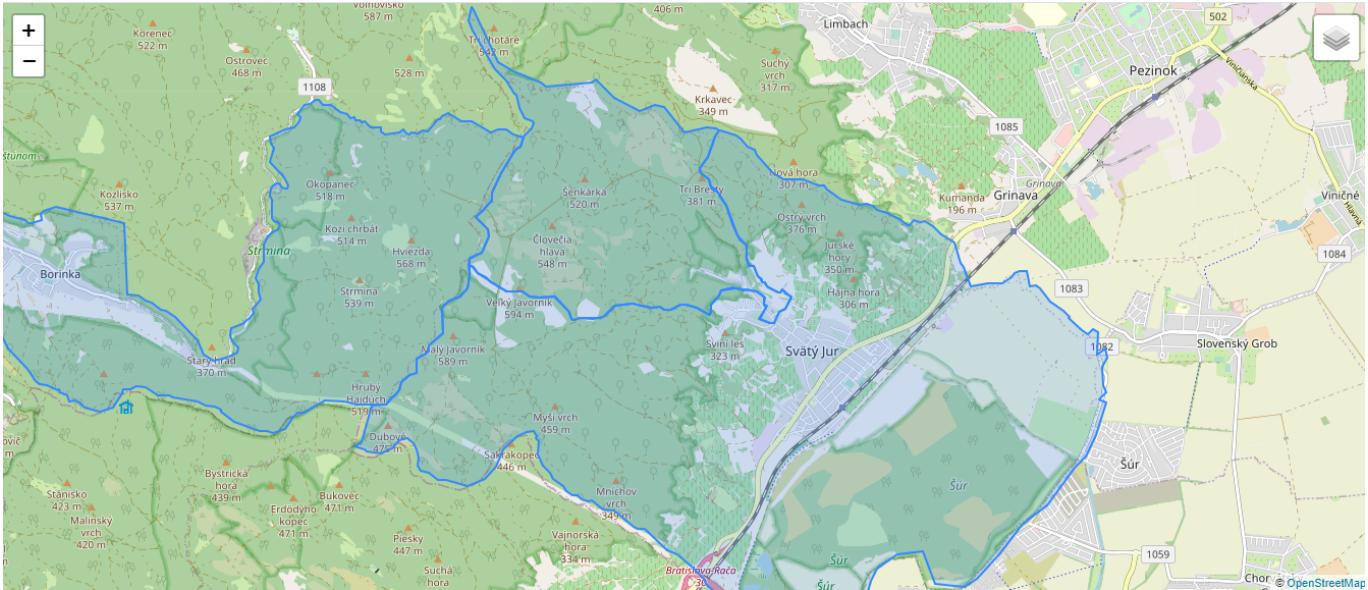
```

1 with road as(
2 with region as(
3 SELECT name, way FROM planet_osm_polygon
4 join (select ST_ASTEXT(way) as house_way FROM planet_osm_polygon WHERE name = 'house') as house on ST_intersects(house.house_way, ST_ASTEXT(planet_osm_poly
5 where planet_osm_polygon.admin_level = '8'
6 )
7 SELECT roads.way FROM planet_osm_roads as roads
8 join region on ST_CONTAINS(region.way, roads.way) ORDER BY roads.way DESC limit 1
9 )
10 SELECT idn5 as id, nm5 as name, ST_TRANSFORM(ST_FORCE2D(kuku."Shape") ,4326)::geography, ST_TRANSFORM(ST_FORCE2D(road.way) ,4326)::geography
11 FROM ku_0 as kuku
12 join road on ST_INTERSECTS(road.way, ST_TRANSFORM(ST_FORCE2D(kuku."Shape") ,3857)::geometry);

```

Data Output		Explain	Messages	Notifications	Geometry Viewer
	id integer	name character varying (50)			st_transform geography
1	803693	Borinka	0106000020E6100000...		0102000020E6100000...
2	822884	Svätý Jur	0106000020E6100000...		0102000020E6100000...
3	822892	Neštich	0106000020E6100000...		0102000020E6100000...





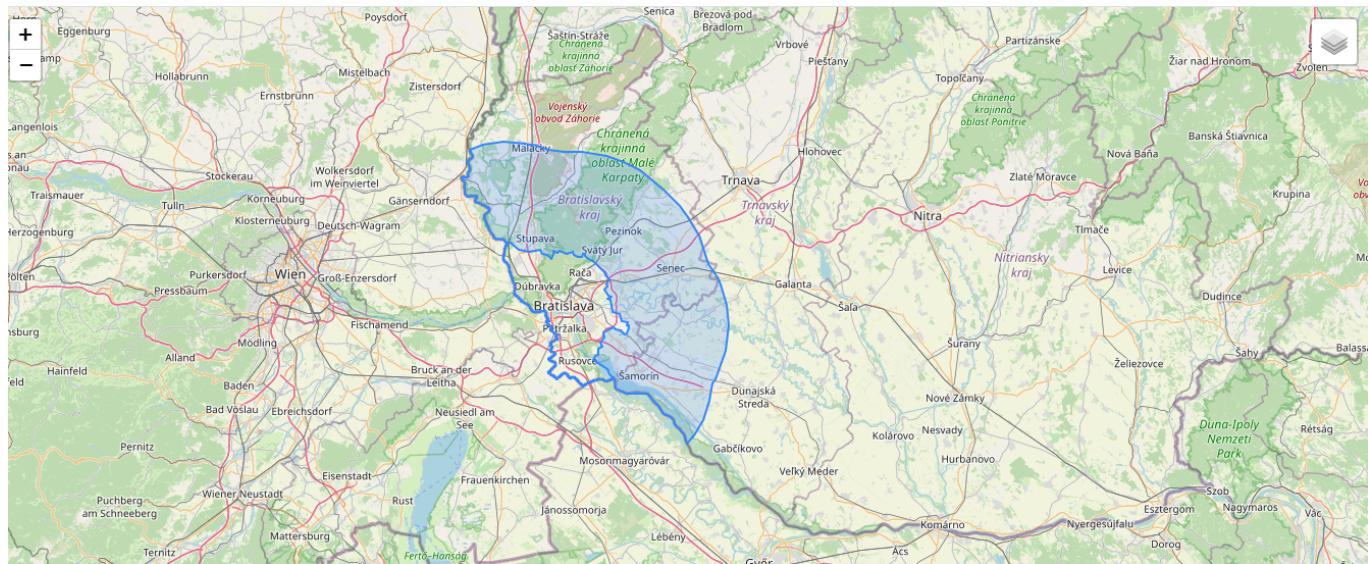
V tejto úlohe som si najskôr vyselektoval okres v ktorom sa nachádza môj dom pomocou funkcie ST_INTERSECT (dom bude mať jeden prienik s okresom). Ďalej som našiel najdlhšiu cestu v danom okrese a to pomocou funkcie CONTAINS (pretože najdlhší výsek cesty môže začínať a končiť na okrajoch okresu čo by nepokryl WITHIN). Potom som vyselektoval katastrálne územia, ktoré majú prienik s danou cestou. Používal som opäť geografiu pre presnejšie výsledky.

```
with road as(
with region as(
SELECT name, way FROM planet_osm_polygon
join (select ST_ASTEXT(way) as house_way FROM planet_osm_polygon WHERE name =
'house') as house on ST_intersects(house.house_way,
ST_ASTEXT(planet_osm_polygon.way))
where planet_osm_polygon.admin_level = '8'
)
SELECT roads.way FROM planet_osm_roads as roads
join region on ST_CONTAINS(region.way, roads.way) ORDER BY roads.way DESC limit 1
)
SELECT idn5 as id, nm5 as name,
ST_TRANSFORM(ST_FORCE2D(kuku."Shape"),4326)::geography,
ST_TRANSFORM(ST_FORCE2D(road.way),4326)::geography
FROM ku_0 as kuku
join road on ST_INTERSECTS(road.way,
ST_TRANSFORM(ST_FORCE2D(kuku."Shape"),3857)::geometry);
```

Uloha 13

Zadanie: Vytvorte oblasť Okolie_Bratislavu, ktorá bude zahŕňať zónu do 20 km od Bratislavu, ale nebude zahŕňať oblasť Bratislavu (Bratislava I až Bratislava V) a bude len na území Slovenska. Zistite jej výmeru.

	area double precision	st_transform geography
1	1480.0760922749641	0107000020E6100000...



Najskôr som vyselektoval Bratislavu z planet_osm_polygon tak, že som selektol všetky okresy 1-5 a pomocou ST_UNION ich spojil do jedného polygónu (admin_level 8 a názov okres Bratislava%). Potom som pomocou ST_Buffer našiel plochu v okolí 20km od Bratislavы, kde som opäť využil geografické údaje pre lepšiu presnosť. Napríklad pre geometriu plocha nepokryla ani Senec čo je podiaľnici 14km. Pomocou SymDifference som vyselektoval len plochu mimo Bratislavу. Nasledne prienik so Slovenskom odstránil všetky časti ležiaci mimo Slovensko. Plochu som získal pomocou ST_Area funkcie. Vyselektoval som si aj plochu v geografii pre znázornenie na obrázku.

```

with BA_area as (
with Bratislava_okolie as(
with Bratislava as(
SELECT ST_UNION(way) as way FROM planet_osm_polygon WHERE admin_level = '8' and
name like ('okres Bratislava%')
)
SELECT
ST_SymDifference((ST_Transform(ST_GeomFromText(ST_ASTEXT(ST_BUFFER(ST_TRANSFORM(Br
atislava.way,4326)::geography, 20000)), 4326), 3857)::geometry), Bratislava.way)
as way
FROM Bratislava
)
SELECT ST_Intersection((Select Bratislava_okolie.way From Bratislava_okolie),
SVK.way) as area
FROM planet_osm_polygon as SVK
WHERE name = 'Slovensko' and admin_level = '2'
)
SELECT ST_AREA(ST_TRANSFORM(BA_area.area, 4326)::geography)/10^6 as area,
ST_TRANSFORM(BA_area.area, 4326)::geography FROM BA_area;

```