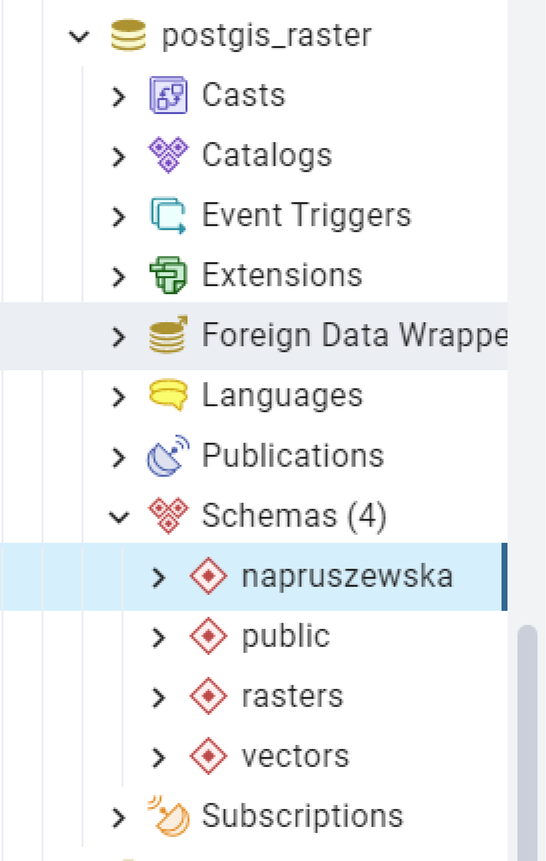
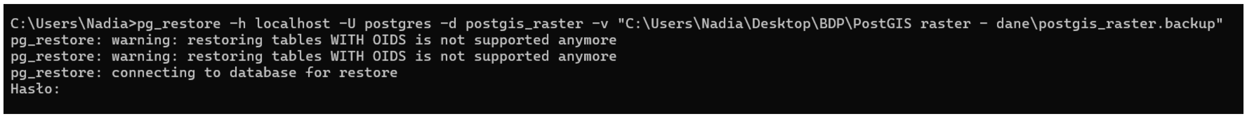
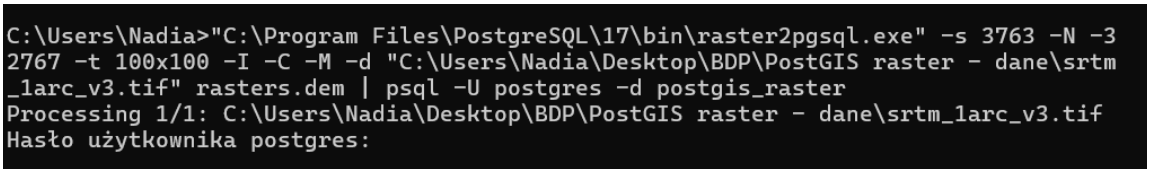
Załadowanie struktury danych



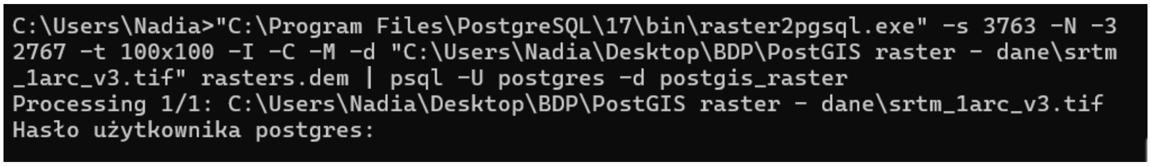


```
C:\Users\Nadia>pg_restore -h localhost -U postgres -d postgis_raster -v "C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\postgis_raster.backup"
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
pg_restore: connecting to database for restore
Hasło:
```

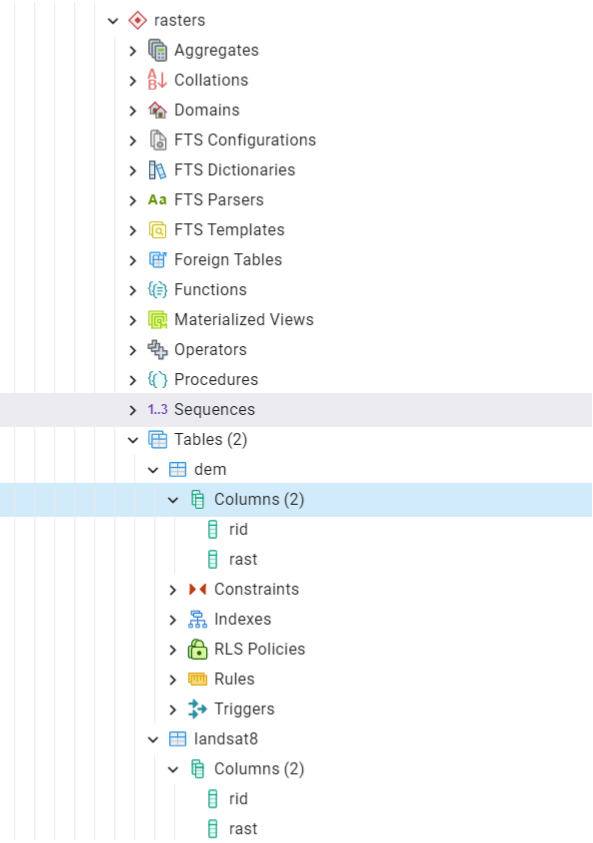
1. Załadowanie rastrów



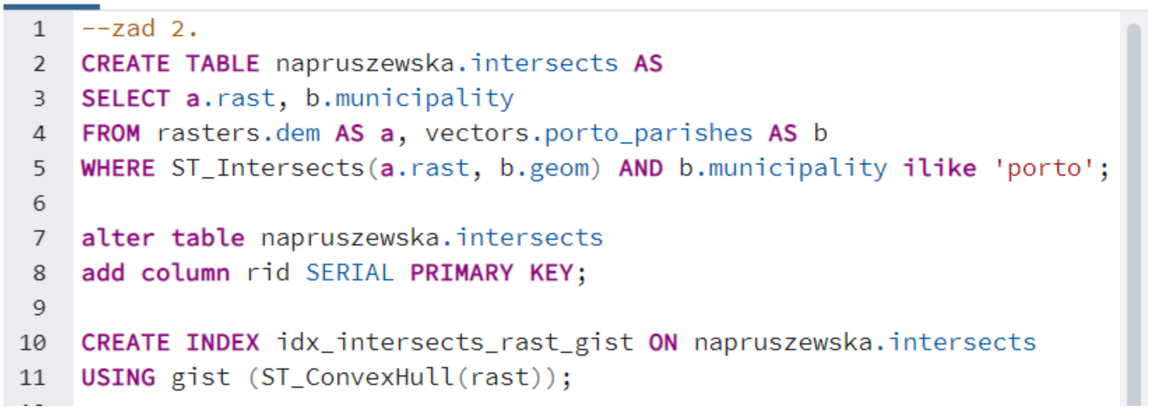
```
C:\Users\Nadia>"C:\Program Files\PostgreSQL\17\bin\raster2pgsql.exe" -s 3763 -N -3
2767 -t 100x100 -I -C -M -d "C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\srtm
_1arc_v3.tif" rasters.dem | psql -U postgres -d postgis_raster
Processing 1/1: C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\srtm_1arc_v3.tif
Hasło użytkownika postgres:
```



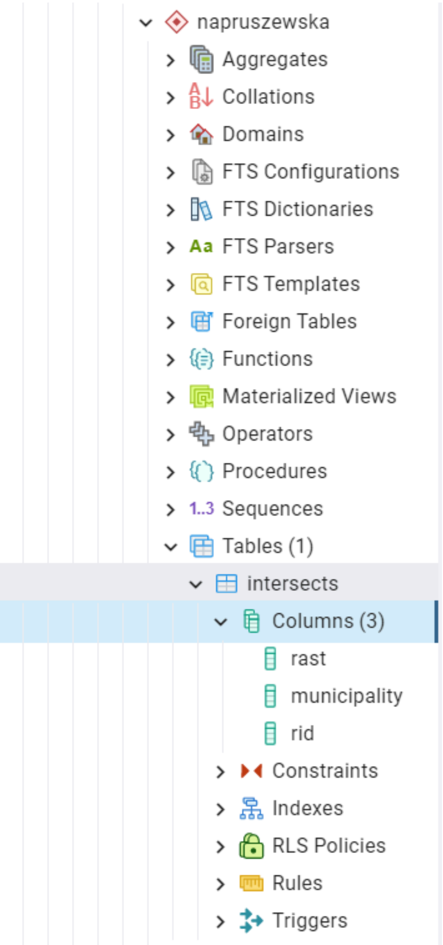
```
C:\Users\Nadia>"C:\Program Files\PostgreSQL\17\bin\raster2pgsql.exe" -s 3763 -N -3
2767 -t 100x100 -I -C -M -d "C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\srtm
_1arc_v3.tif" rasters.dem | psql -U postgres -d postgis_raster
Processing 1/1: C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\srtm_1arc_v3.tif
Hasło użytkownika postgres:
```



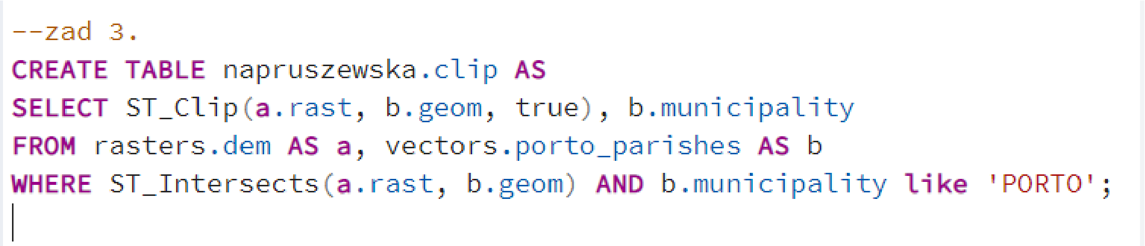
2. ST_Intersects



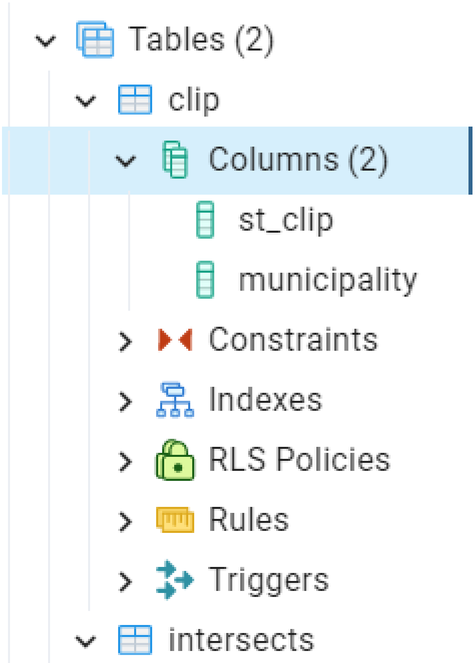
```
--zad 2.
CREATE TABLE napruszewska.intersects AS
SELECT a.rast, b.municipality
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto';

alter table napruszewska.intersects
add column rid SERIAL PRIMARY KEY;

CREATE INDEX idx_intersects_rast_gist ON napruszewska.intersects
USING gist (ST_ConvexHull(rast));
```



3. ST_Clip

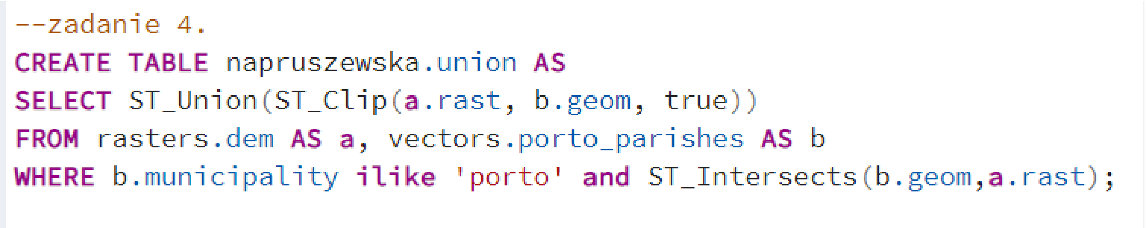


```
--zad 3.
CREATE TABLE napruszewska.clip AS
SELECT ST_Clip(a.rast, b.geom, true), b.municipality
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE ST_Intersects(a.rast, b.geom) AND b.municipality like 'PORTO';
```

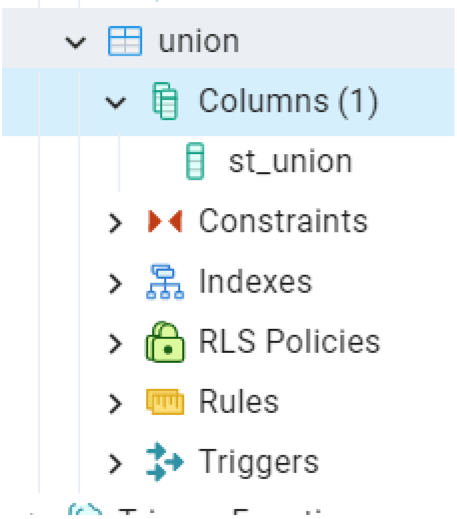


Tables (2)
clip
Columns (2)
st_clip
municipality
Constraints
Indexes
RLS Policies
Rules
Triggers
intersects

4. ST_Union

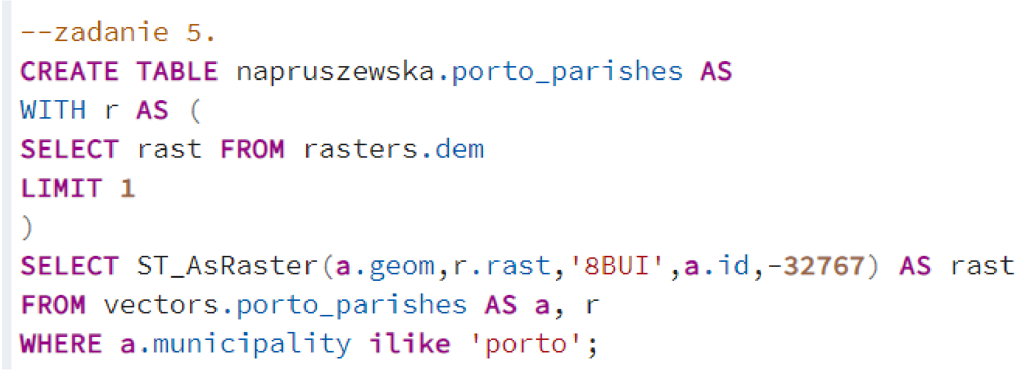


```
--zadanie 4.
CREATE TABLE napruszewska.union AS
SELECT ST_Union(ST_Clip(a.rast, b.geom, true))
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast);
```

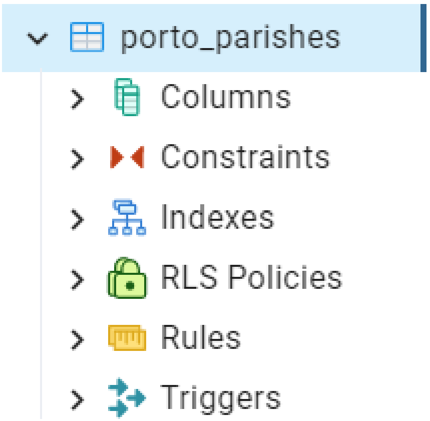


union
Columns (1)
st_union
Constraints
Indexes
RLS Policies
Rules
Triggers

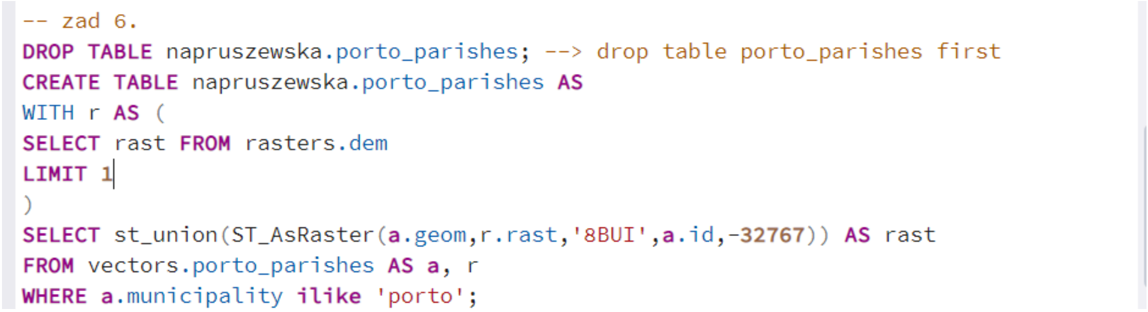
5. ST_AsRaster



```
--zadanie 5.
CREATE TABLE napruszewska.porto_parishes AS
WITH r AS (
SELECT rast FROM rasters.dem
LIMIT 1
)
SELECT ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality ilike 'porto';
```

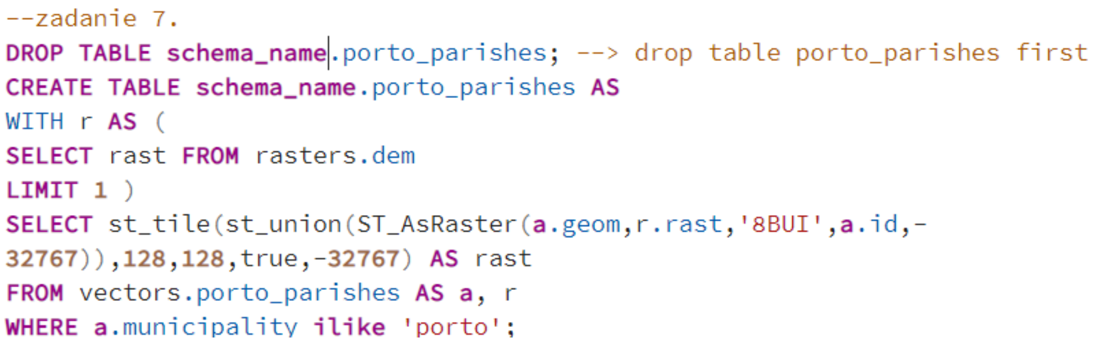


6. St_union



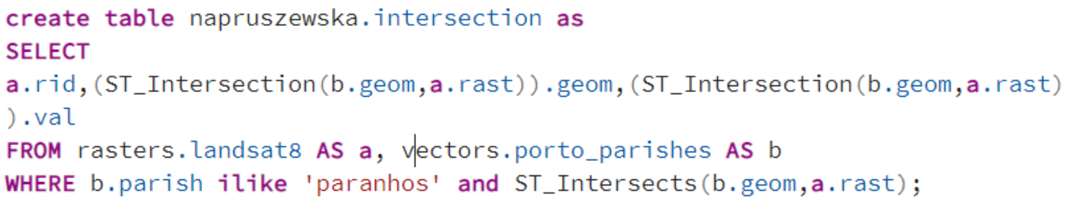
```
-- zad 6.
DROP TABLE napruszewska.porto_parishes; --> drop table porto_parishes first
CREATE TABLE napruszewska.porto_parishes AS
WITH r AS (
SELECT rast FROM rasters.dem
LIMIT 1
)
SELECT st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767)) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality ilike 'porto';
```

7. ST_Tile



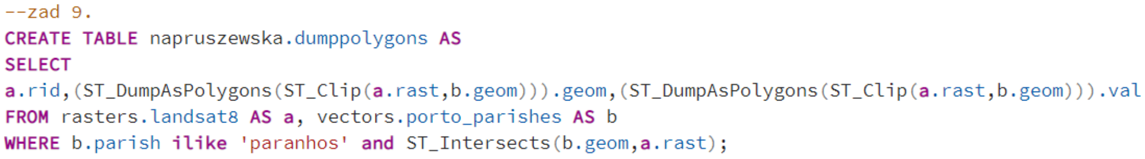
```
--zadanie 7.
DROP TABLE schema_name.porto_parishes; --> drop table porto_parishes first
CREATE TABLE schema_name.porto_parishes AS
WITH r AS (
SELECT rast FROM rasters.dem
LIMIT 1 )
SELECT st_tile(st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-
32767)),128,128,true,-32767) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality ilike 'porto';
```

8. ST_Intersection



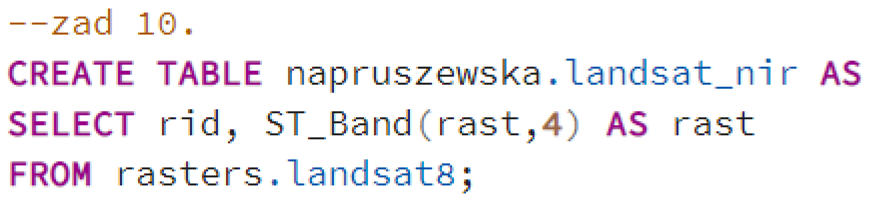
```
create table napruszewska.intersection as
SELECT
a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast)
).val
FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

9. ST_DumpAsPolygons



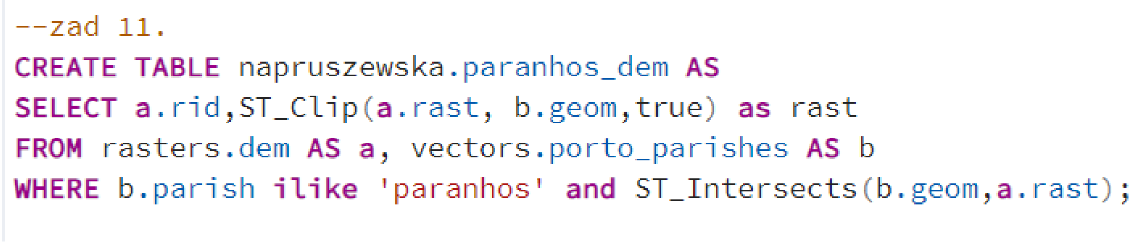
```
--zad 9.
CREATE TABLE napruszewska.dumppolygons AS
SELECT
a.rid,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).geom,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).val
FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

10. ST_Band



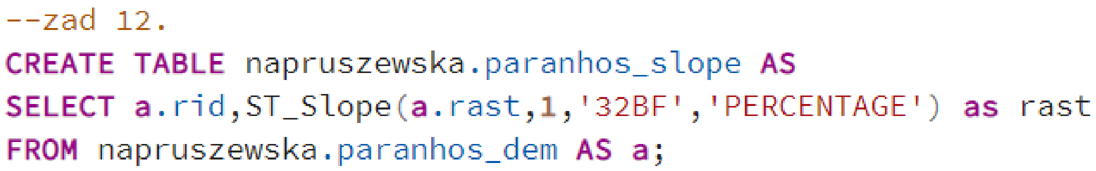
```
--zad 10.
CREATE TABLE napruszewska.landsat_nir AS
SELECT rid, ST_Band(rast,4) AS rast
FROM rasters.landsat8;
```

11. ST_Clip



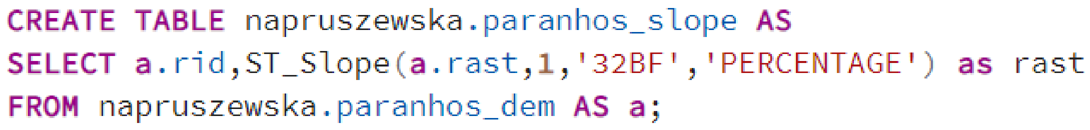
```
--zad 11.
CREATE TABLE napruszewska.paranhos_dem AS
SELECT a.rid,ST_Clip(a.rast, b.geom,true) as rast
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

12. - ST_Slope



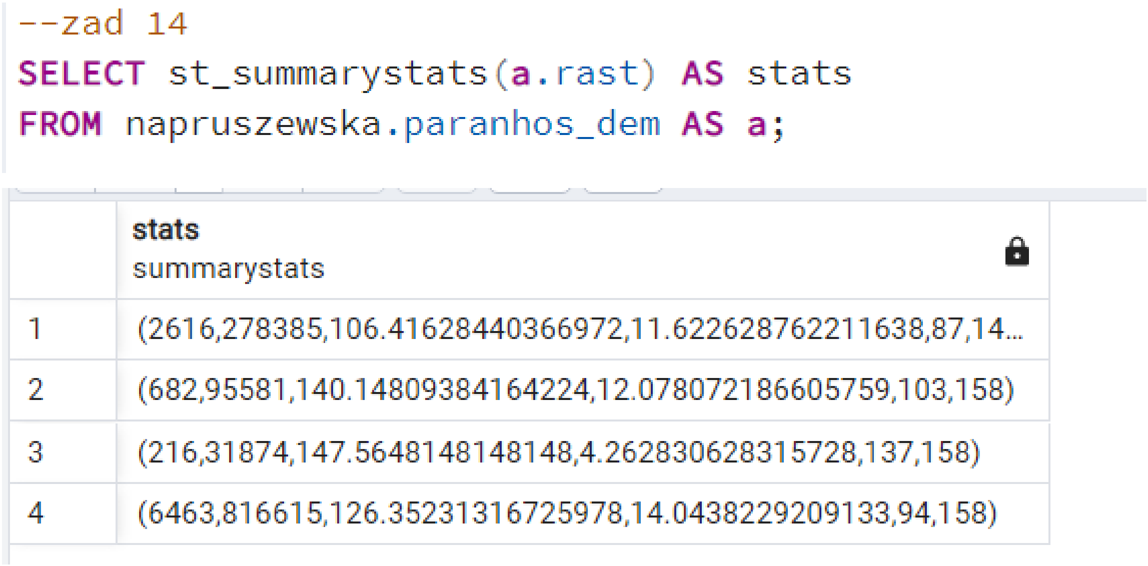
```
--zad 12.
CREATE TABLE napruszewska.paranhos_slope AS
SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
FROM napruszewska.paranhos_dem AS a;
```

13. ST_Reclass



```
CREATE TABLE napruszewska.paranhos_slope AS
SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
FROM napruszewska.paranhos_dem AS a;
```

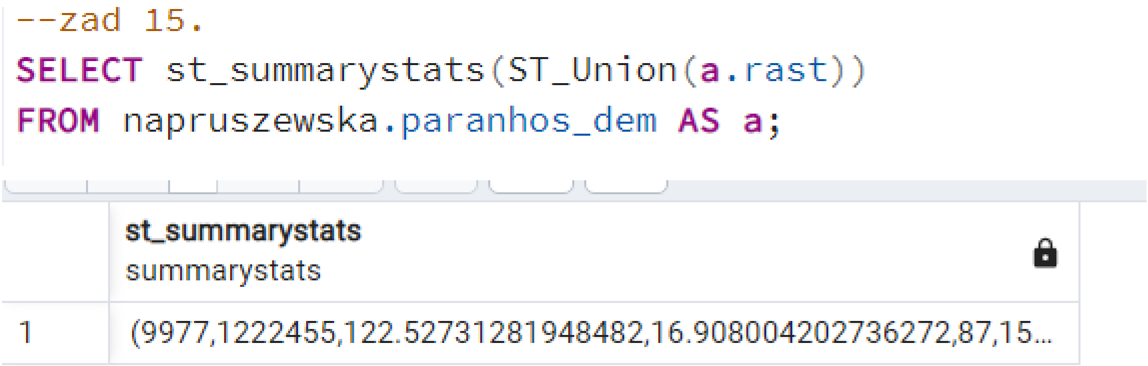
14. ST_SummaryStats



```
--zad 14
SELECT st_summarystats(a.rast) AS stats
FROM napruszewska.paranhos_dem AS a;
```

	stats summarystats
1	(2616,278385,106.41628440366972,11.622628762211638,87,14...
2	(682,95581,140.14809384164224,12.078072186605759,103,158)
3	(216,31874,147.5648148148148,4.262830628315728,137,158)
4	(6463,816615,126.35231316725978,14.0438229209133,94,158)

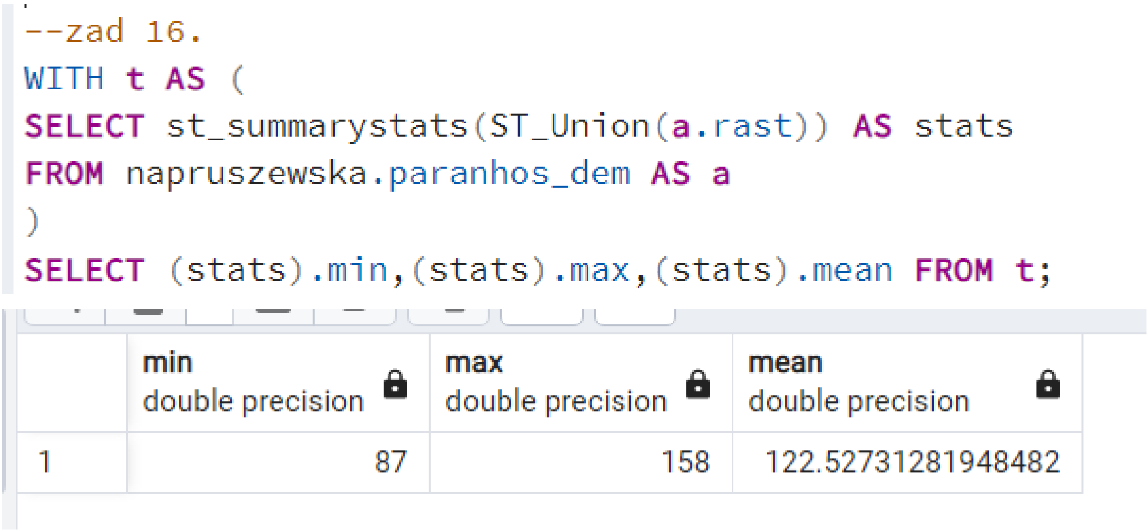
15. ST_SummaryStats oraz Union



```
--zad 15.
SELECT st_summarystats(ST_Union(a.rast))
FROM napruszewska.paranhos_dem AS a;
```

	st_summarystats summarystats
1	(9977,1222455,122.52731281948482,16.908004202736272,87,15...

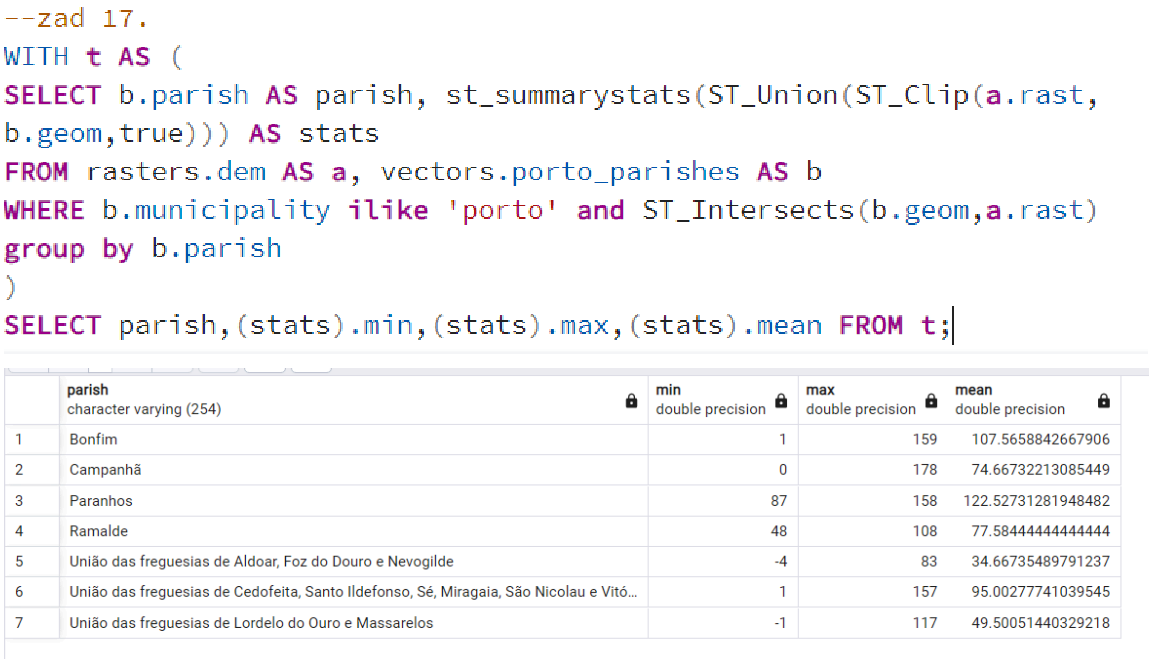
16. ST_SummaryStats z lepszą kontrolą złożonego typu danych



```
--zad 16.
WITH t AS (
SELECT st_summarystats(ST_Union(a.rast)) AS stats
FROM napruszewska.paranhos_dem AS a
)
SELECT (stats).min,(stats).max,(stats).mean FROM t;
```

	min double precision	max double precision	mean double precision
1	87	158	122.52731281948482

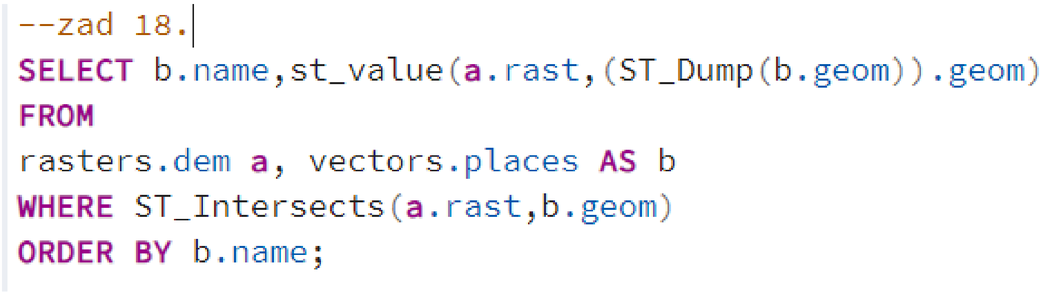
17. - ST_SummaryStats w połączeniu z GROUP BY



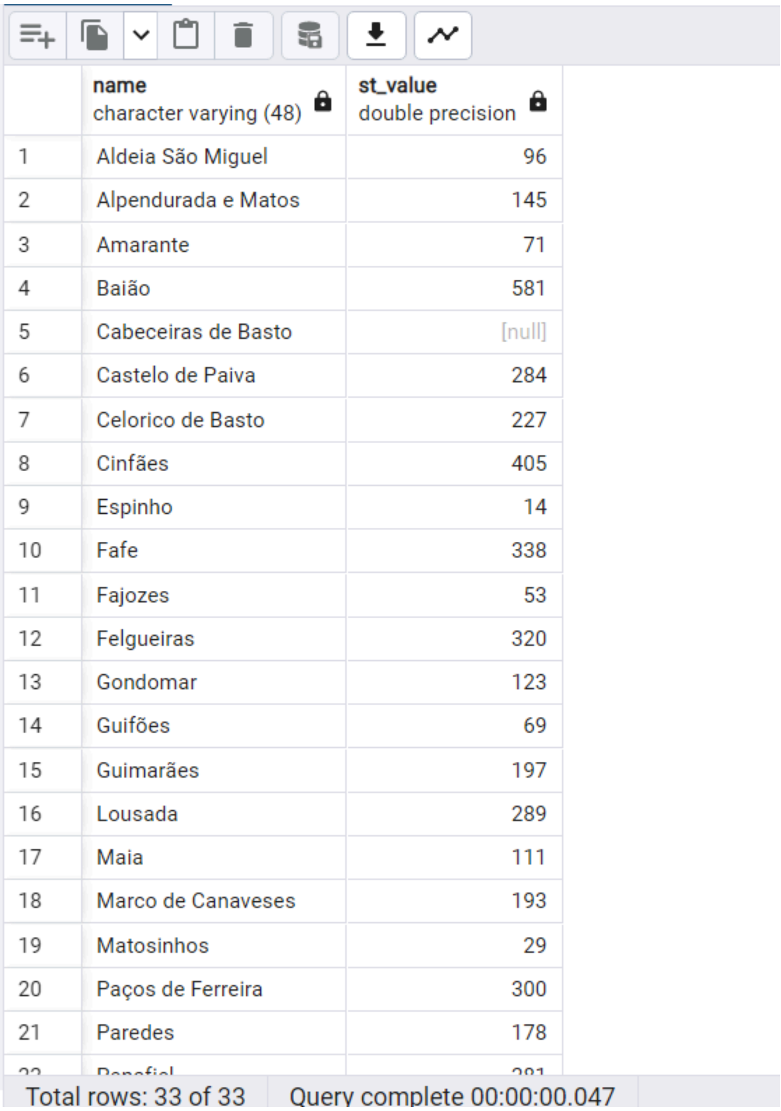
```
--zad 17.
WITH t AS (
SELECT b.parish AS parish, st_summarystats(ST_Union(ST_Clip(a.rast,
b.geom,true))) AS stats
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
group by b.parish
)
SELECT parish,(stats).min,(stats).max,(stats).mean FROM t;
```

	parish character varying (254)	min double precision	max double precision	mean double precision
1	Bonfim	1	159	107.5658842667906
2	Campanhã	0	178	74.66732213085449
3	Paranhos	87	158	122.52731281948482
4	Ramalde	48	108	77.58444444444444
5	União das freguesias de Aldoar, Foz do Douro e Nevogilde	-4	83	34.66735489791237
6	União das freguesias de Cedofeita, Santo Ildefonso, Sé, Miragaia, São Nicolau e Vitó...	1	157	95.00277741039545
7	União das freguesias de Lordelo do Ouro e Massarelos	-1	117	49.50051440329218

18. ST_Value



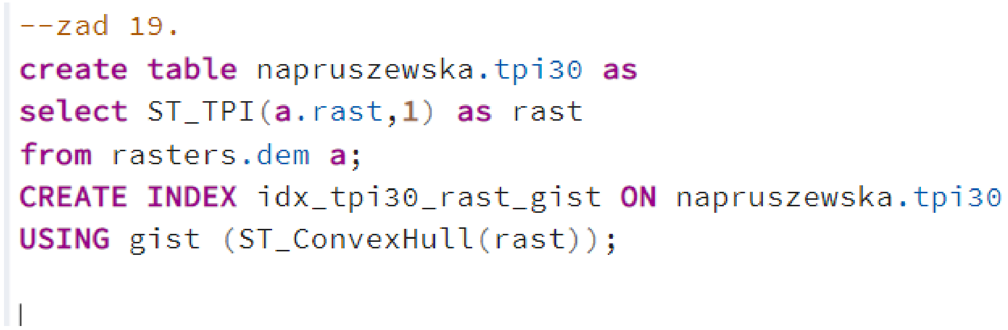
```
--zad 18.
SELECT b.name,st_value(a.rast,(ST_Dump(b.geom)).geom)
FROM
rasters.dem a, vectors.places AS b
WHERE ST_Intersects(a.rast,b.geom)
ORDER BY b.name;
```



	name character varying (48)	st_value double precision
1	Aldeia São Miguel	96
2	Alpendurada e Matos	145
3	Amarante	71
4	Baião	581
5	Cabeceiras de Basto	[null]
6	Castelo de Paiva	284
7	Celorico de Basto	227
8	Cinfães	405
9	Espinho	14
10	Fafe	338
11	Fajozes	53
12	Felgueiras	320
13	Gondomar	123
14	Guifões	69
15	Guimarães	197
16	Lousada	289
17	Maia	111
18	Marco de Canaveses	193
19	Matosinhos	29
20	Paços de Ferreira	300
21	Paredes	178
22	Penafiel	281

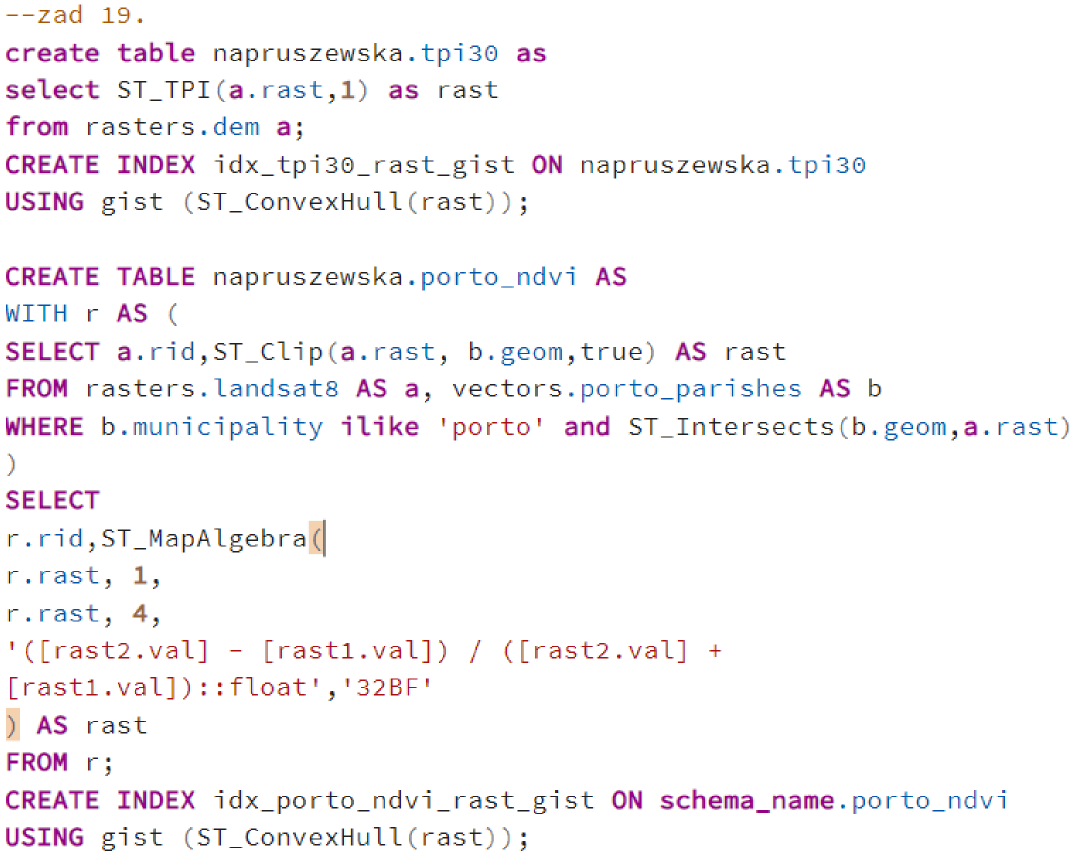
Total rows: 33 of 33   Query complete 00:00:00.047

19. ST_TPI



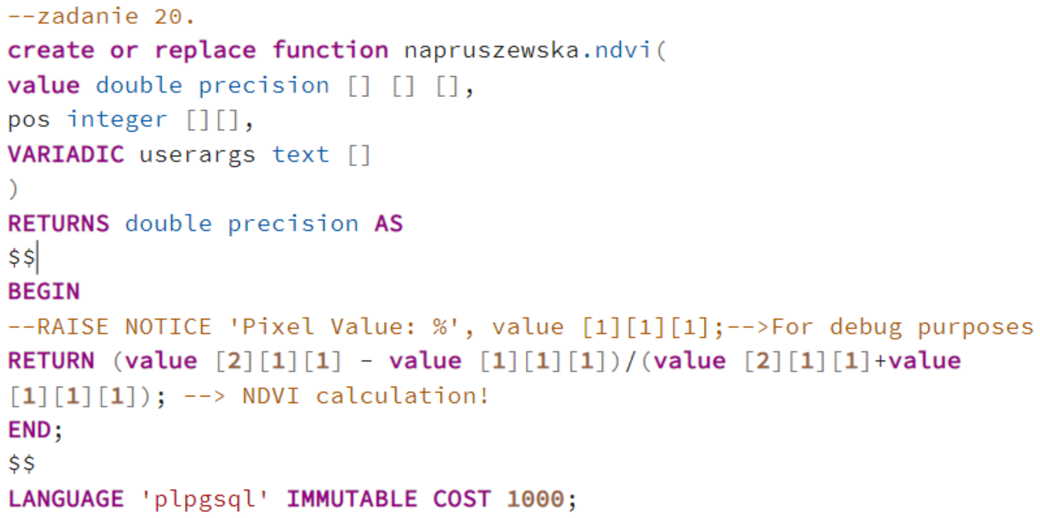
```
--zad 19.
create table napruszewska.tpi30 as
select ST_TPI(a.rast,1) as rast
from rasters.dem a;
CREATE INDEX idx_tpi30_rast_gist ON napruszewska.tpi30
USING gist (ST_ConvexHull(rast));
```

20. Wyrażenie Algebry Map

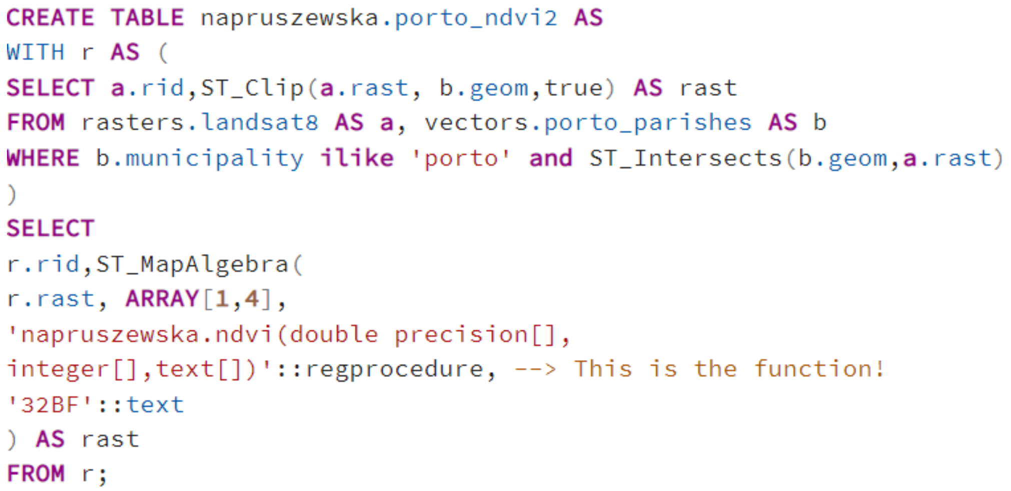


```
--zad 19.
create table napruszewska.tpi30 as
select ST_TPI(a.rast,1) as rast
from rasters.dem a;
CREATE INDEX idx_tpi30_rast_gist ON napruszewska.tpi30
USING gist (ST_ConvexHull(rast));

CREATE TABLE napruszewska.porto_ndvi AS
WITH r AS (
SELECT a.rid,ST_Clip(a.rast, b.geom,true) AS rast
FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
)
SELECT
r.rid,ST_MapAlgebra(
r.rast, 1,
r.rast, 4,
'([rast2.val] - [rast1.val]) / ([rast2.val] +
[rast1.val])::float','32BF'
) AS rast
FROM r;
CREATE INDEX idx_porto_ndvi_rast_gist ON schema_name.porto_ndvi
USING gist (ST_ConvexHull(rast));
```

21. Funkcja zwrotna



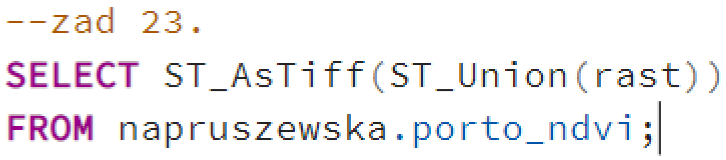
```
--zadanie 20.
create or replace function napruszewska.ndvi(
value double precision [] [] [],
pos integer [][],
VARIADIC userargs text []
)
RETURNS double precision AS
$$
BEGIN
--RAISE NOTICE 'Pixel Value: %', value [1][1][1];-->For debug purposes
RETURN (value [2][1][1] - value [1][1][1])/(value [2][1][1]+value
[1][1][1]); --> NDVI calculation!
END;
$$
LANGUAGE 'plpgsql' IMMUTABLE COST 1000;
```



```
CREATE TABLE napruszewska.porto_ndvi2 AS
WITH r AS (
SELECT a.rid,ST_Clip(a.rast, b.geom,true) AS rast
FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
)
SELECT
r.rid,ST_MapAlgebra(
r.rast, ARRAY[1,4],
'napruszewska.ndvi(double precision[],
integer[],text[])'::regprocedure, --> This is the function!
'32BF'::text
) AS rast
FROM r;
```

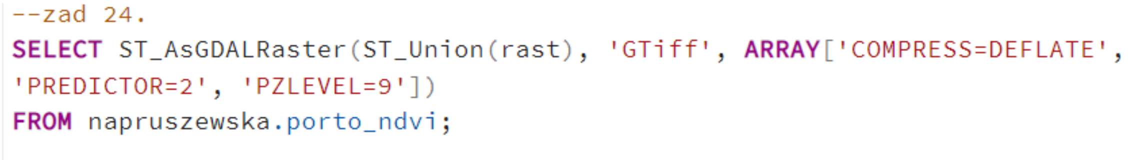
22.

23. ST_AsTiff

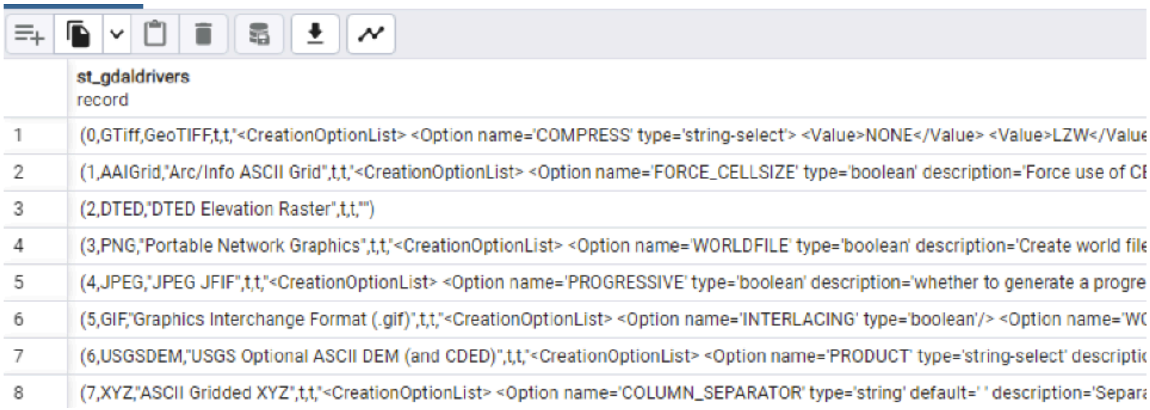


```
--zad 23.
SELECT ST_AsTiff(ST_Union(rast))
FROM napruszewska.porto_ndvi;
```

24. ST_AsGDALRaster

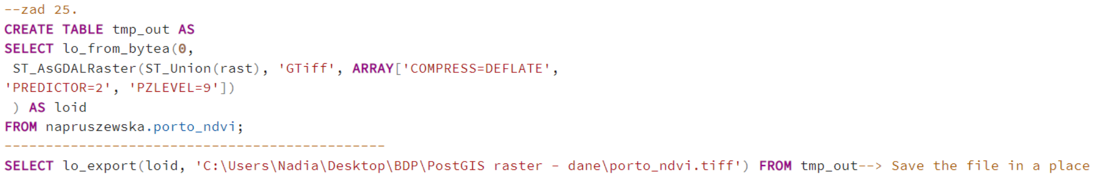


```sql
--zad 24.
SELECT ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE',
'PREDICTOR=2', 'PZLEVEL=9'])
FROM napruszewska.porto_ndvi;
```



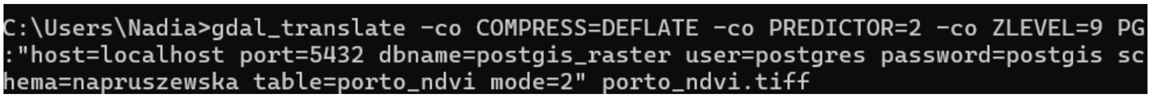
	st_gdaldrivers record
1	(0,GTiff,GeoTIFF,t,t,'<CreationOptionList> <Option name='COMPRESS' type='string-select'> <Value>NONE</Value> <Value>LZW</Value
2	(1,AAIGrid,"Arc/Info ASCII Grid",t,t,'<CreationOptionList> <Option name='FORCE_CELLSIZE' type='boolean' description='Force use of CE
3	(2,DTED,"DTED Elevation Raster",t,t,"")
4	(3,PNG,"Portable Network Graphics",t,t,"<CreationOptionList> <Option name='WORLDFILE' type='boolean' description='Create world file
5	(4,JPEG,"JPEG JFIF",t,t,"<CreationOptionList> <Option name='PROGRESSIVE' type='boolean' description='whether to generate a progre
6	(5,GIF,"Graphics Interchange Format (.gif)",t,t,"<CreationOptionList> <Option name='INTERLACING' type='boolean'/> <Option name='WC
7	(6,USGSDEM,"USGS Optional ASCII DEM (and CDED)",t,t,"<CreationOptionList> <Option name='PRODUCT' type='string-select' descriptio
8	(7,XYZ,"ASCII Gridded XYZ",t,t,"<CreationOptionList> <Option name='COLUMN_SEPARATOR' type='string' default=' ' description='Separa

25. Zapisywanie danych na dysku za pomocą dużego obiektu (large object, lo)



```sql
--zad 25.
CREATE TABLE tmp_out AS
SELECT lo_from_bytea(0,
 ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE',
'PREDICTOR=2', 'PZLEVEL=9'])
 ) AS loid
FROM napruszewska.porto_ndvi;
----------------------------------------------
SELECT lo_export(loid, 'C:\Users\Nadia\Desktop\BDP\PostGIS raster - dane\porto_ndvi.tiff') FROM tmp_out--> Save the file in a place
```

26. Użycie GDAL



```
C:\Users\Nadia>gdal_translate -co COMPRESS=DEFLATE -co PREDICTOR=2 -co ZLEVEL=9 PG
:"host=localhost port=5432 dbname=postgis_raster user=postgres password=postgis sc
hema=napruszewska table=porto_ndvi mode=2" porto_ndvi.tiff
```