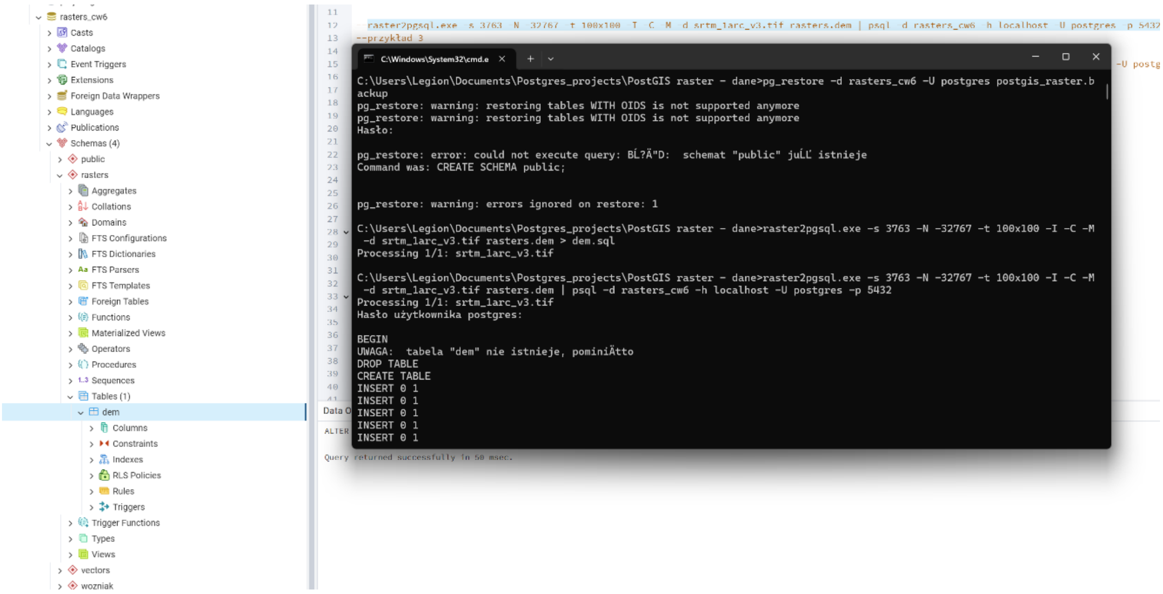
Ćwiczenia 6-7 Rasters, Bazy Danych Przestrzennych

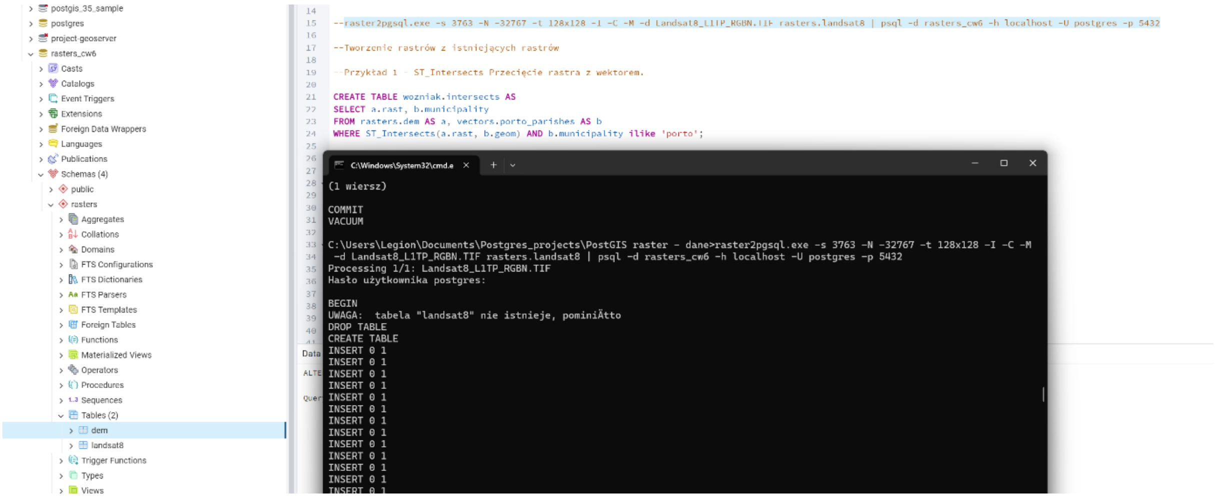
Krystian Woźniak



```
--raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d srtm_1arc_v3.tif rasters.dem | psql -d rasters_cw6 -h localhost -U postgres -p 5432
--przykład 3
```

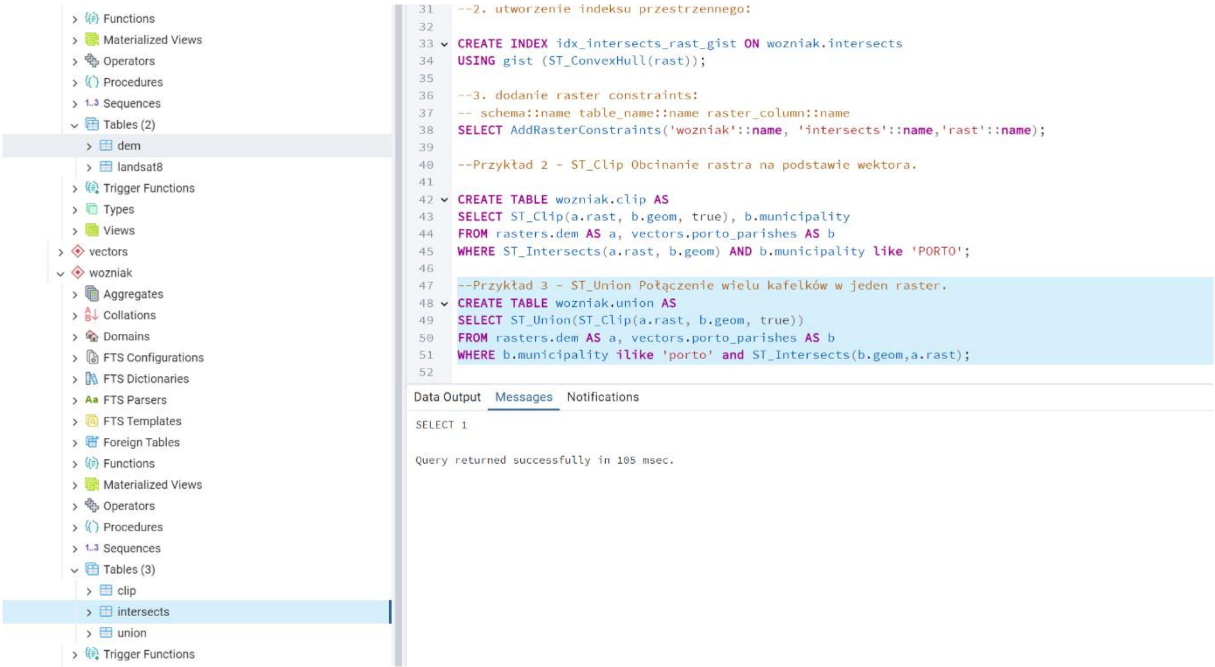
```
C:\Users\Legion\Documents\Postgres_projects\PostGIS raster - dane>pg_restore -d rasters_cw6 -U postgres postgis_raster.b
ackup
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
Hasło:

pg_restore: error: could not execute query: BĹ?Ä"D:  schemat "public" juĹĽ istnieje
Command was: CREATE SCHEMA public;


pg_restore: warning: errors ignored on restore: 1

C:\Users\Legion\Documents\Postgres_projects\PostGIS raster - dane>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M
 -d srtm_1arc_v3.tif rasters.dem > dem.sql
Processing 1/1: srtm_1arc_v3.tif

C:\Users\Legion\Documents\Postgres_projects\PostGIS raster - dane>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M
 -d srtm_1arc_v3.tif rasters.dem | psql -d rasters_cw6 -h localhost -U postgres -p 5432
Processing 1/1: srtm_1arc_v3.tif
Hasło użytkownika postgres:

BEGIN
UWAGA:  tabela "dem" nie istnieje, pominiĂtto
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
```

```
Query returned successfully in 50 msec.
```



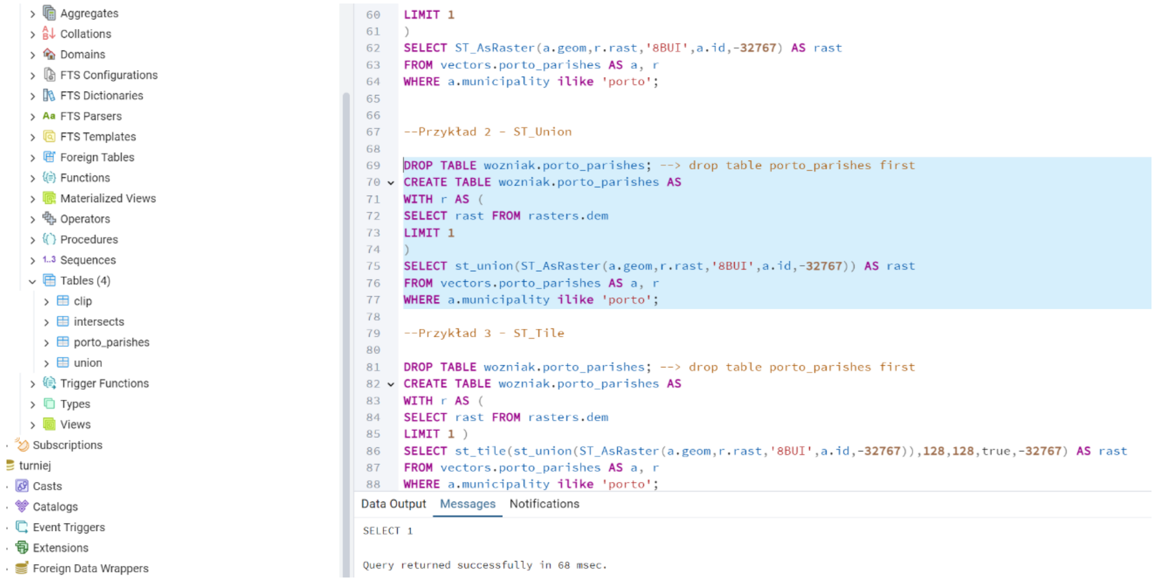
```
--raster2pgsql.exe -s 3763 -N -32767 -t 128x128 -I -C -M -d Landsat8_L1TP_RGBN.TIF rasters.landsat8 | psql -d rasters_cw6 -h localhost -U postgres -p 5432

--Tworzenie rastrów z istniejących rastrów

--Przykład 1 - ST_Intersects Przecięcie rastra z wektorem.

CREATE TABLE wozniak.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';
```

```
(1 wiersz)

COMMIT
VACUUM

C:\Users\Legion\Documents\Postgres_projects\PostGIS raster - dane>raster2pgsql.exe -s 3763 -N -32767 -t 128x128 -I -C -M
 -d Landsat8_L1TP_RGBN.TIF rasters.landsat8 | psql -d rasters_cw6 -h localhost -U postgres -p 5432
Processing 1/1: Landsat8_L1TP_RGBN.TIF
Hasło użytkownika postgres:

BEGIN
UWAGA:  tabela "landsat8" nie istnieje, pominiĂtto
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
```



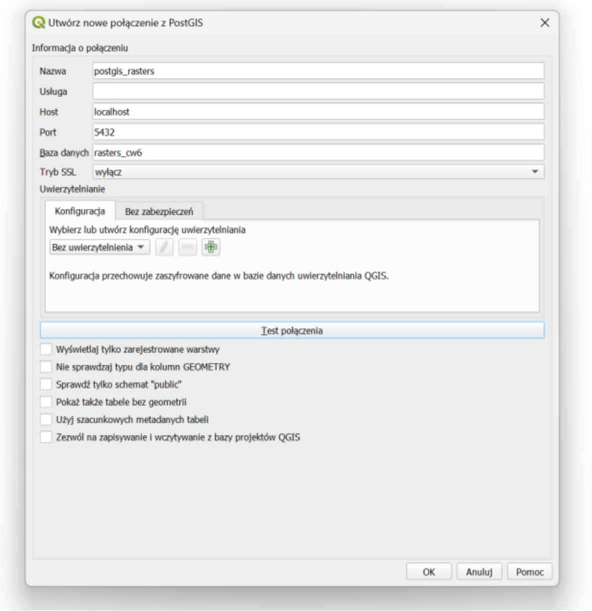
```
--2. utworzenie indeksu przestrzennego:

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

--3. dodanie raster constraints:
-- schema::name table_name::name raster_column::name
SELECT AddRasterConstraints('wozniak'::name, 'intersects'::name,'rast'::name);

--Przyktad 2 - ST_Clip Obcinanie rastra na podstawie wektora.

CREATE TABLE wozniak.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';

--Przyktad 3 - ST_Union Połączenie wielu kafelków w jeden raster.
CREATE TABLE wozniak.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);
```

```
SELECT 1

Query returned successfully in 105 msec.
```



```
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';


--Przykład 2 - ST_Union

DROP TABLE wozniak.porto_parishes; --> drop table porto_parishes first
CREATE TABLE wozniak.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';

--Przykład 3 - ST_Tile

DROP TABLE wozniak.porto_parishes; --> drop table porto_parishes first
CREATE TABLE wozniak.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';
```

```
SELECT 1

Query returned successfully in 68 msec.
```



Utwórz nowe połączenie z PostGIS

Informacja o połączeniu

Nazwa: postgis_rasters

Usługa:

Host: localhost

Port: 5432

Baza danych: rasters_cw6

Tryb SSL: wyłącz

Uwierzytelnianie

Wybierz lub utwórz konfigurację uwierzytelniania

Bez uwierzytelnienia

Konfiguracja przechowuje zaszyfrowane dane w bazie danych uwierzytelniania QGIS.

☐ Wyświetlaj tylko zarejestrowane warstwy

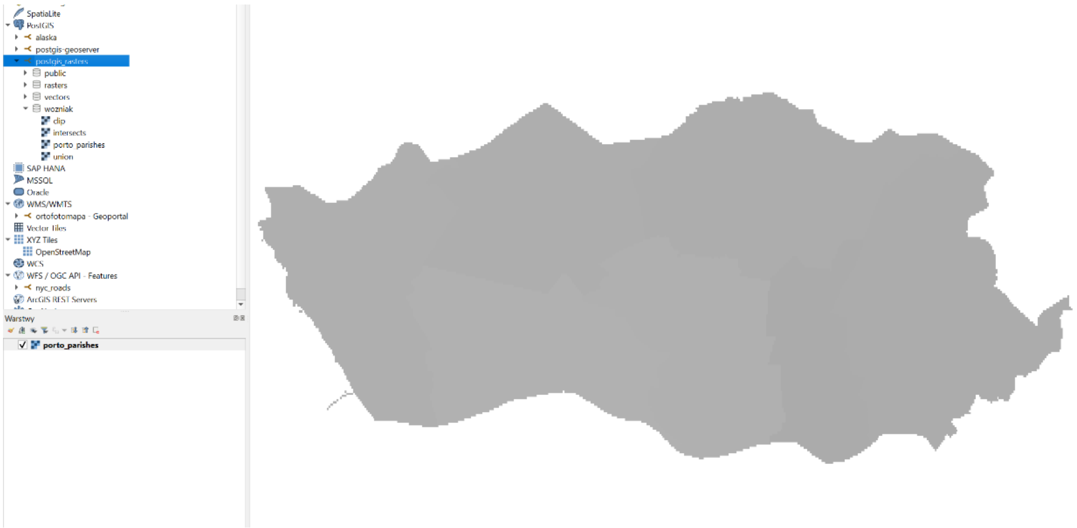
☐ Nie sprawdzaj typu dla kolumn GEOMETRY

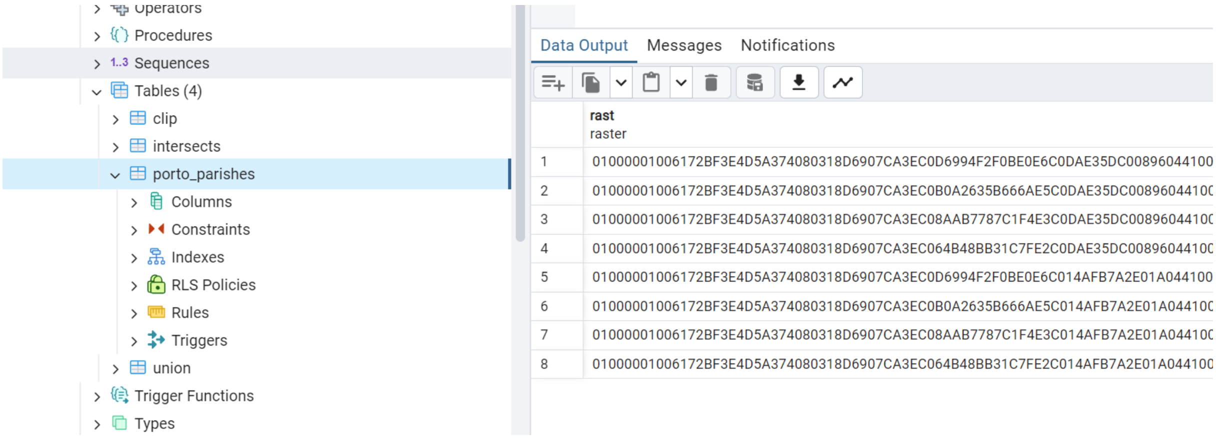
☐ Sprawdź tylko schemat "public"

☐ Pokaż także tabele bez geometrii

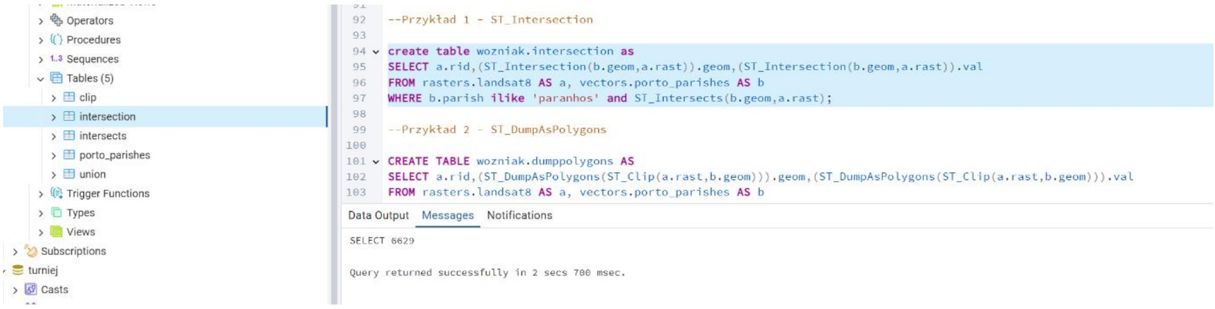
☐ Użyj szacunkowych metadanych tabeli

☐ Zezwól na zapisywanie i wczytywanie z bazy projektów QGIS



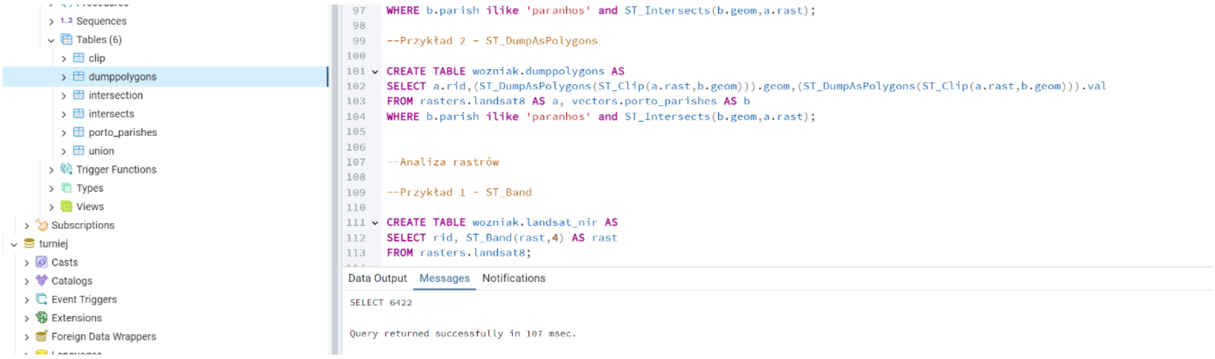


	rast raster
1	01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C0DAE35DC00896044100
2	01000001006172BF3E4D5A374080318D6907CA3EC0B0A2635B666AE5C0DAE35DC0089604410(
3	01000001006172BF3E4D5A374080318D6907CA3EC08AAB7787C1F4E3C0DAE35DC0089604410(
4	01000001006172BF3E4D5A374080318D6907CA3EC064B48BB31C7FE2C0DAE35DC0089604410(
5	01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C014AFB7A2E01A044100
6	01000001006172BF3E4D5A374080318D6907CA3EC0B0A2635B666AE5C014AFB7A2E01A04410(
7	01000001006172BF3E4D5A374080318D6907CA3EC08AAB7787C1F4E3C014AFB7A2E01A04410(
8	01000001006172BF3E4D5A374080318D6907CA3EC064B48BB31C7FE2C014AFB7A2E01A044100



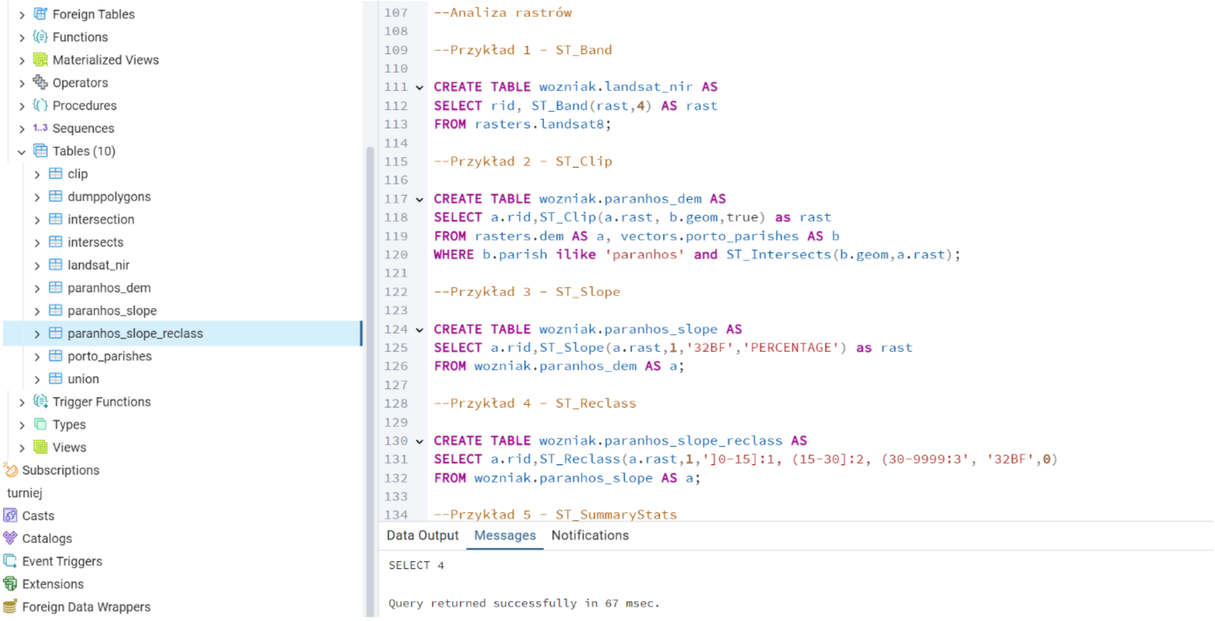
```
--Przykład 1 - ST_Intersection

create table wozniak.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);

--Przykład 2 - ST_DumpAsPolygons

CREATE TABLE wozniak.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
```

```
SELECT 6629

Query returned successfully in 2 secs 700 msec.
```



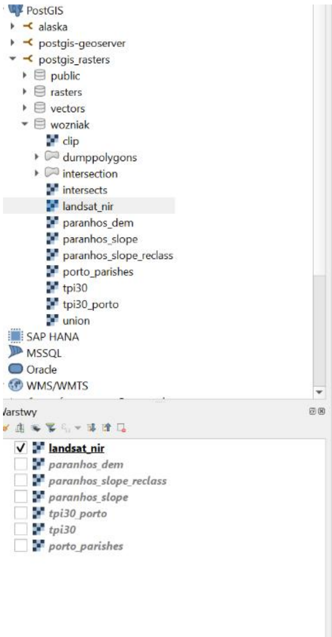
```
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);

--Przykład 2 - ST_DumpAsPolygons

CREATE TABLE wozniak.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);


--Analiza rastrów

--Przykład 1 - ST_Band

CREATE TABLE wozniak.landsat_nir AS
SELECT rid, ST_Band(rast,4) AS rast
FROM rasters.landsat8;
```

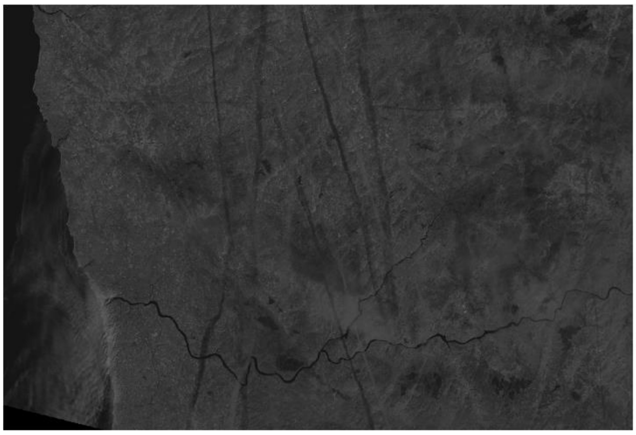
```
SELECT 6422

Query returned successfully in 107 msec.
```

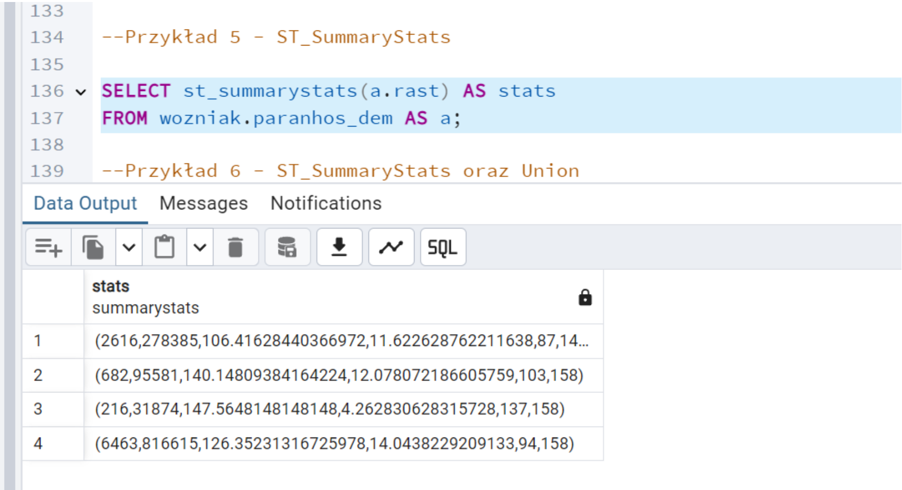


```
--Analiza rastrów

--Przykład 1 - ST_Band

CREATE TABLE wozniak.landsat_nir AS
SELECT rid, ST_Band(rast,4) AS rast
FROM rasters.landsat8;

--Przykład 2 - ST_Clip

CREATE TABLE wozniak.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);

--Przykład 3 - ST_Slope

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

--Przykład 4 - ST_Reclass

CREATE TABLE wozniak.paranhos_slope_reclass AS
SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3', '32BF',0)
FROM wozniak.paranhos_slope AS a;

--Przykład 5 - ST_SummaryStats
```

```
SELECT 4

Query returned successfully in 67 msec.
```

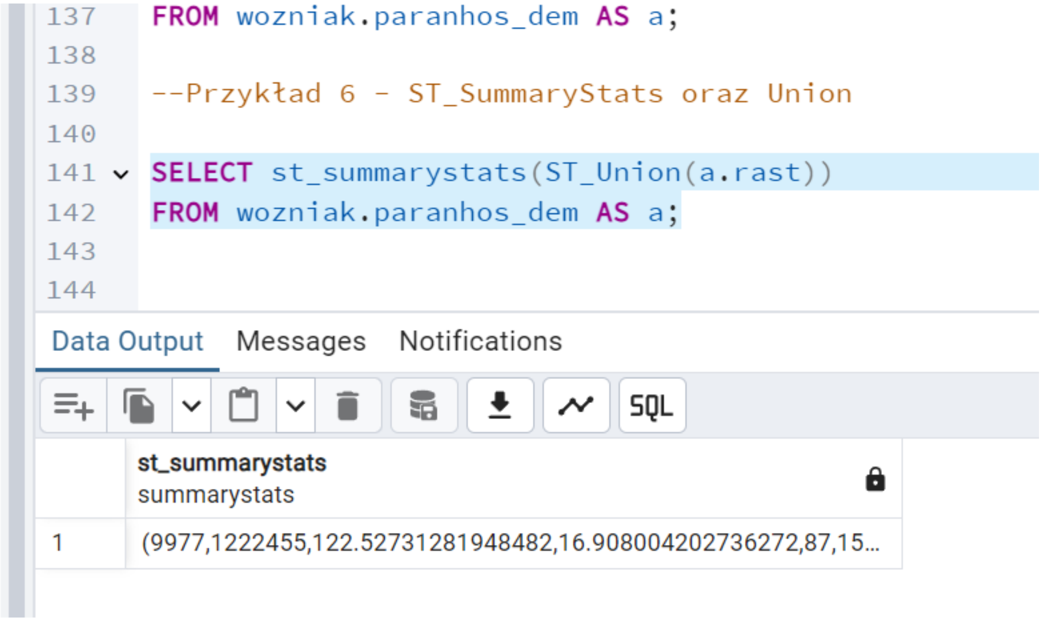






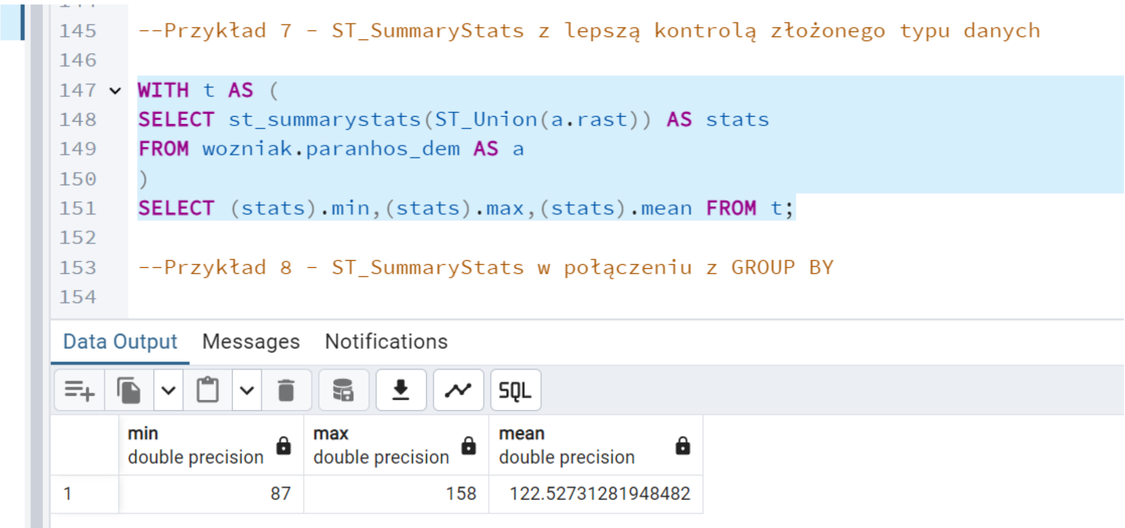
```sql
--Przykład 5 - ST_SummaryStats

SELECT st_summarystats(a.rast) AS stats
FROM wozniak.paranhos_dem AS a;

--Przykład 6 - ST_SummaryStats oraz Union
```

	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)



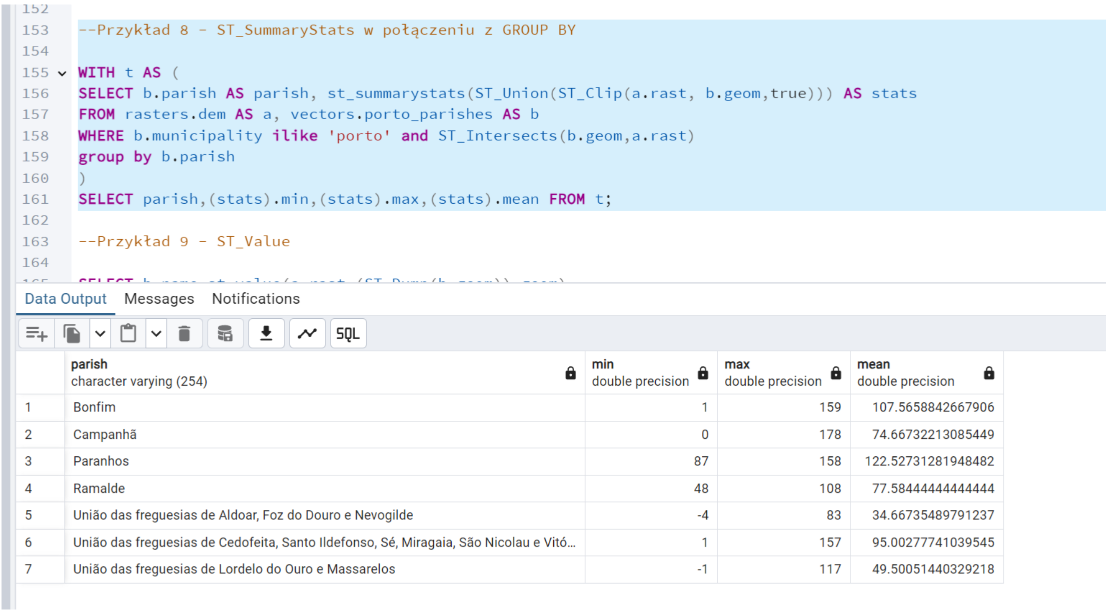
```sql
FROM wozniak.paranhos_dem AS a;

--Przykład 6 - ST_SummaryStats oraz Union

SELECT st_summarystats(ST_Union(a.rast))
FROM wozniak.paranhos_dem AS a;


```

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



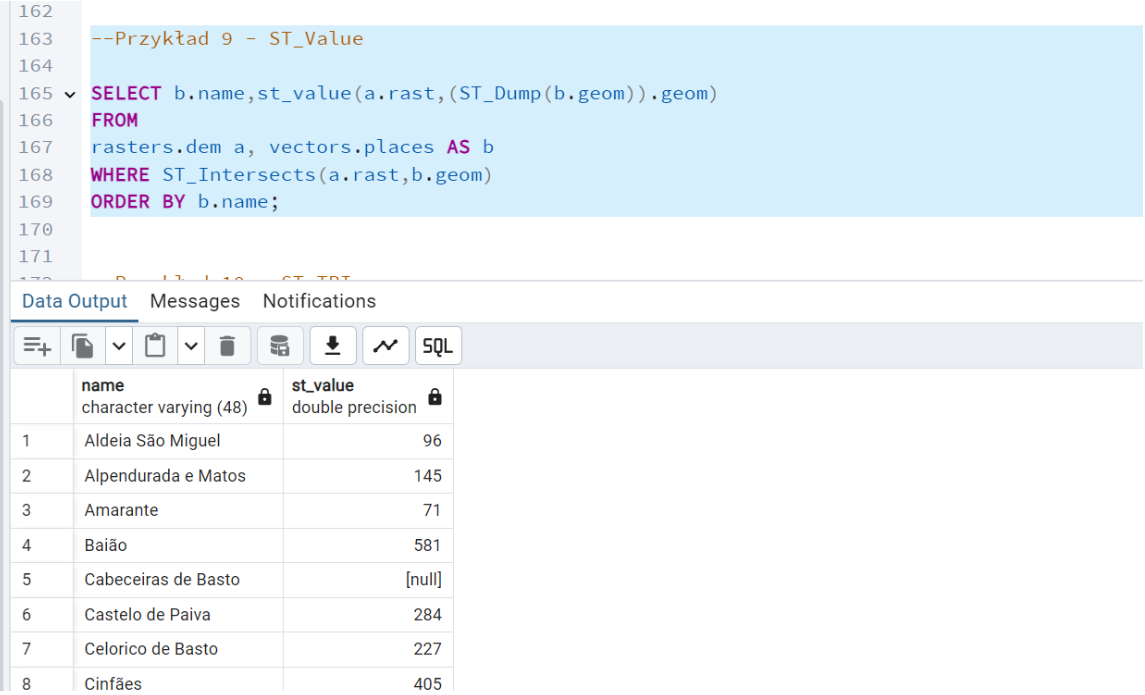
```sql
--Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych

WITH t AS (
SELECT st_summarystats(ST_Union(a.rast)) AS stats
FROM wozniak.paranhos_dem AS a
)
SELECT (stats).min,(stats).max,(stats).mean FROM t;

--Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY
```

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



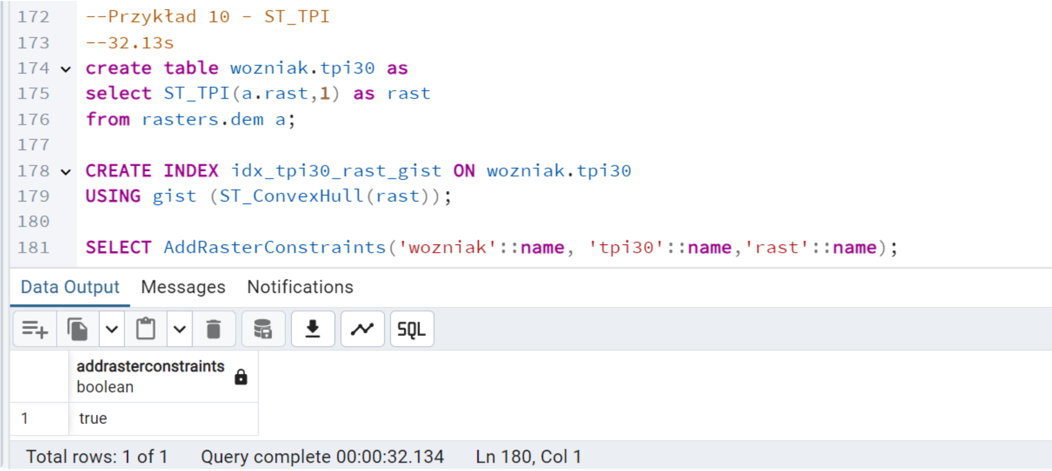
```sql
--Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY

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;

--Przykład 9 - ST_Value
```

	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



```sql
--Przykład 9 - ST_Value

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;
```

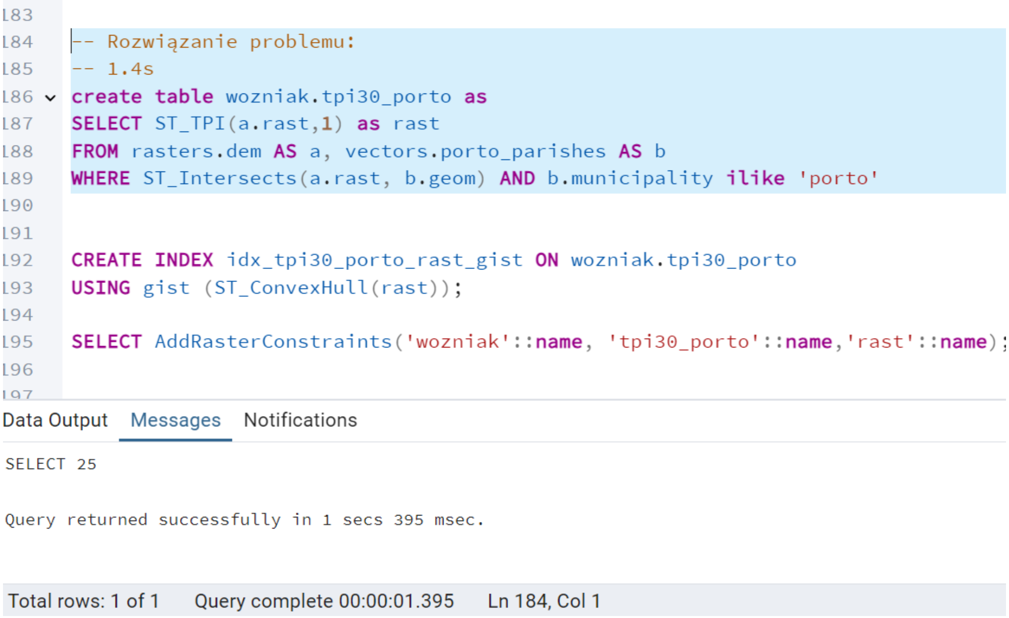
	name character varying (48)	st_value double precision
1	Aldeia São Miguel	96
2	Alpendurada e Matos	145
3	Amarante	71
4	Baião	581
5	Cabeceiras de Basto	[null]
6	Castelo de Paiva	284
7	Celorico de Basto	227
8	Cinfães	405



```
--Przykład 10 - ST_TPI
--32.13s
create table wozniak.tpi30 as
select ST_TPI(a.rast,1) as rast
from rasters.dem a;

CREATE INDEX idx_tpi30_rast_gist ON wozniak.tpi30
USING gist (ST_ConvexHull(rast));

SELECT AddRasterConstraints('wozniak'::name, 'tpi30'::name,'rast'::name);
```

	addrasterconstraints boolean
1	true

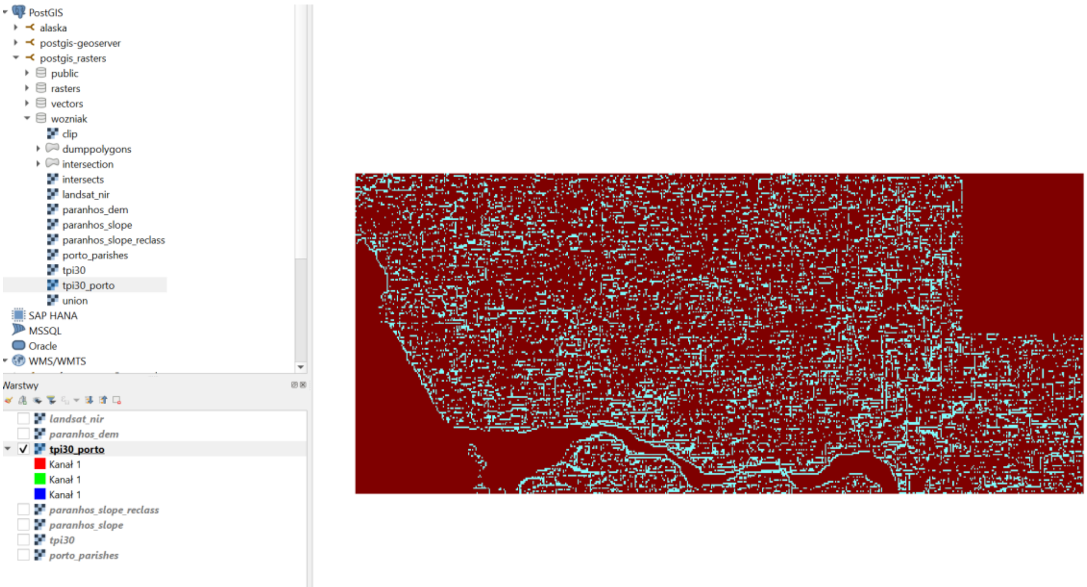
```
Total rows: 1 of 1     Query complete 00:00:32.134     Ln 180, Col 1
```

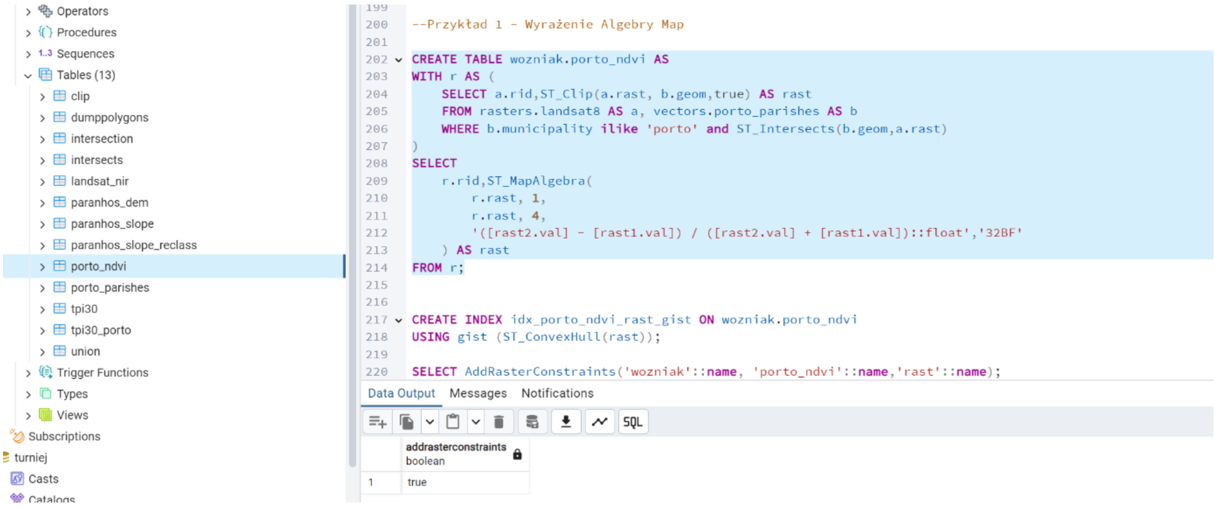


```
-- Rozwiązanie problemu:
-- 1.4s
create table wozniak.tpi30_porto as
SELECT ST_TPI(a.rast,1) as rast
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto'


CREATE INDEX idx_tpi30_porto_rast_gist ON wozniak.tpi30_porto
USING gist (ST_ConvexHull(rast));

SELECT AddRasterConstraints('wozniak'::name, 'tpi30_porto'::name,'rast'::name);
```

```
SELECT 25

Query returned successfully in 1 secs 395 msec.
```

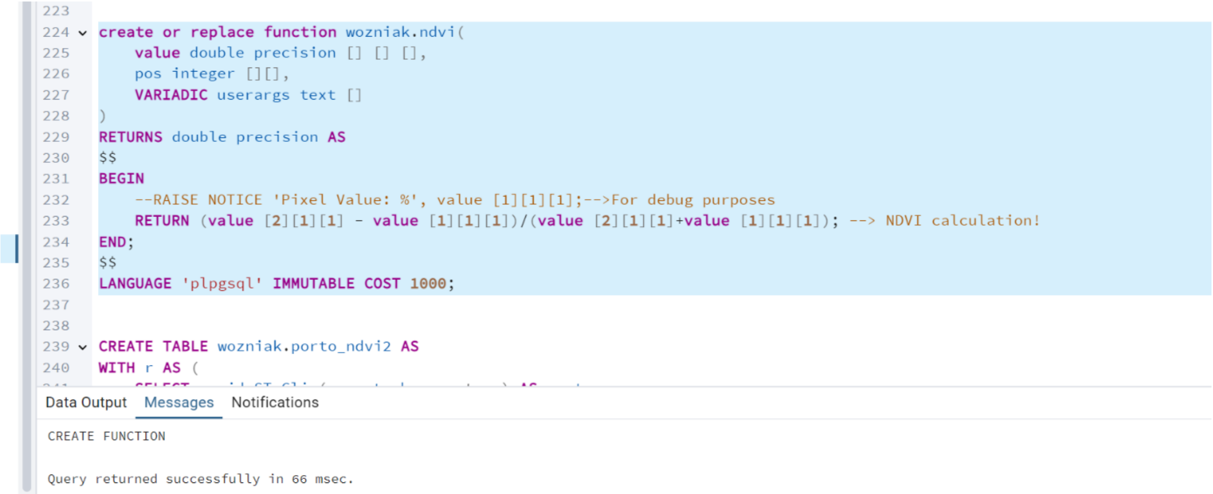
```
Total rows: 1 of 1     Query complete 00:00:01.395     Ln 184, Col 1
```





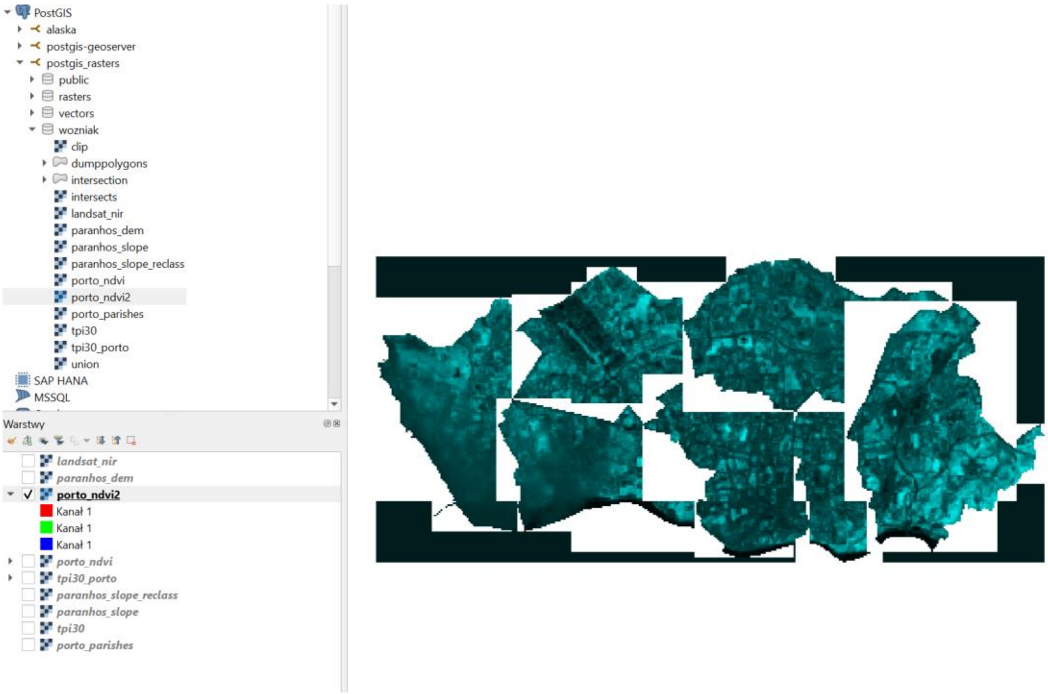
```sql
--Przykład 1 - Wyrażenie Algebry Map

CREATE TABLE wozniak.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 wozniak.porto_ndvi
USING gist (ST_ConvexHull(rast));

SELECT AddRasterConstraints('wozniak'::name, 'porto_ndvi'::name,'rast'::name);
```

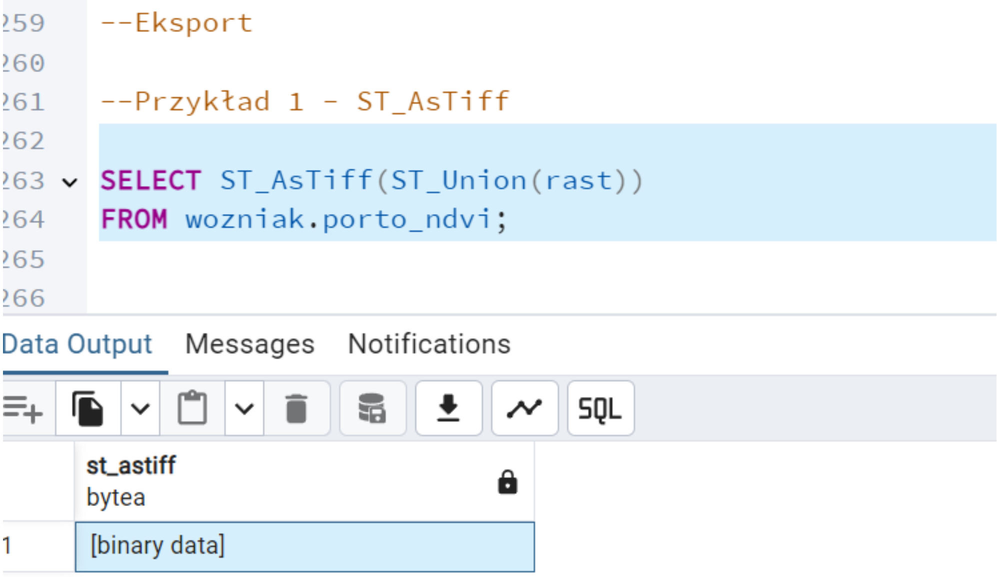
	addrasterconstraints boolean
1	true



```sql
create or replace function wozniak.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 wozniak.porto_ndvi2 AS
WITH r AS (
```

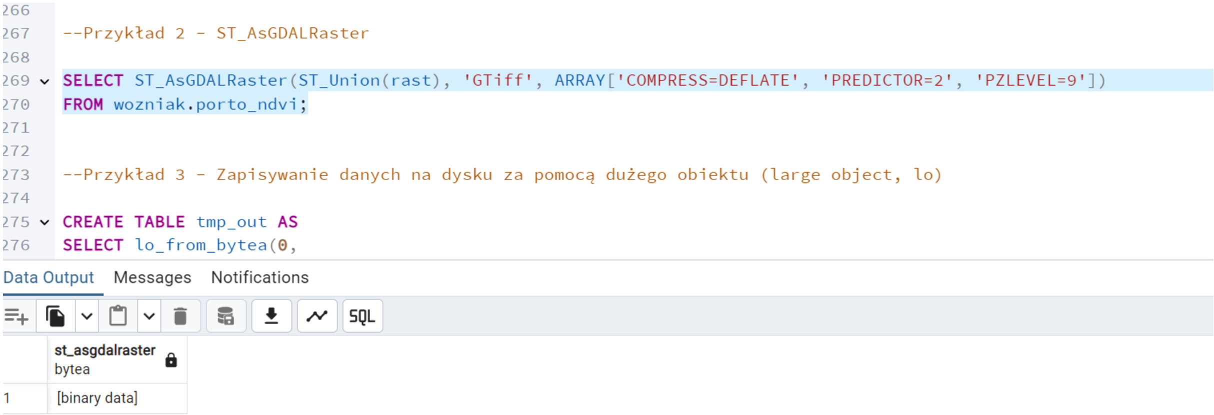
```
CREATE FUNCTION

Query returned successfully in 66 msec.
```





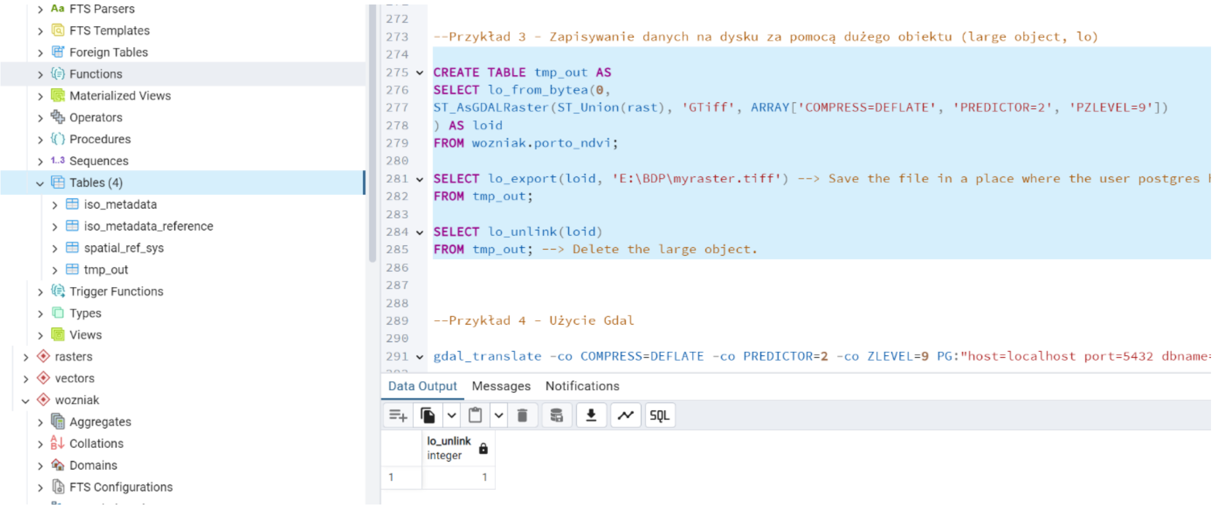
```
--Eksport

--Przykład 1 - ST_AsTiff

SELECT ST_AsTiff(ST_Union(rast))
FROM wozniak.porto_ndvi;
```

```
   st_astiff
   bytea
1  [binary data]
```



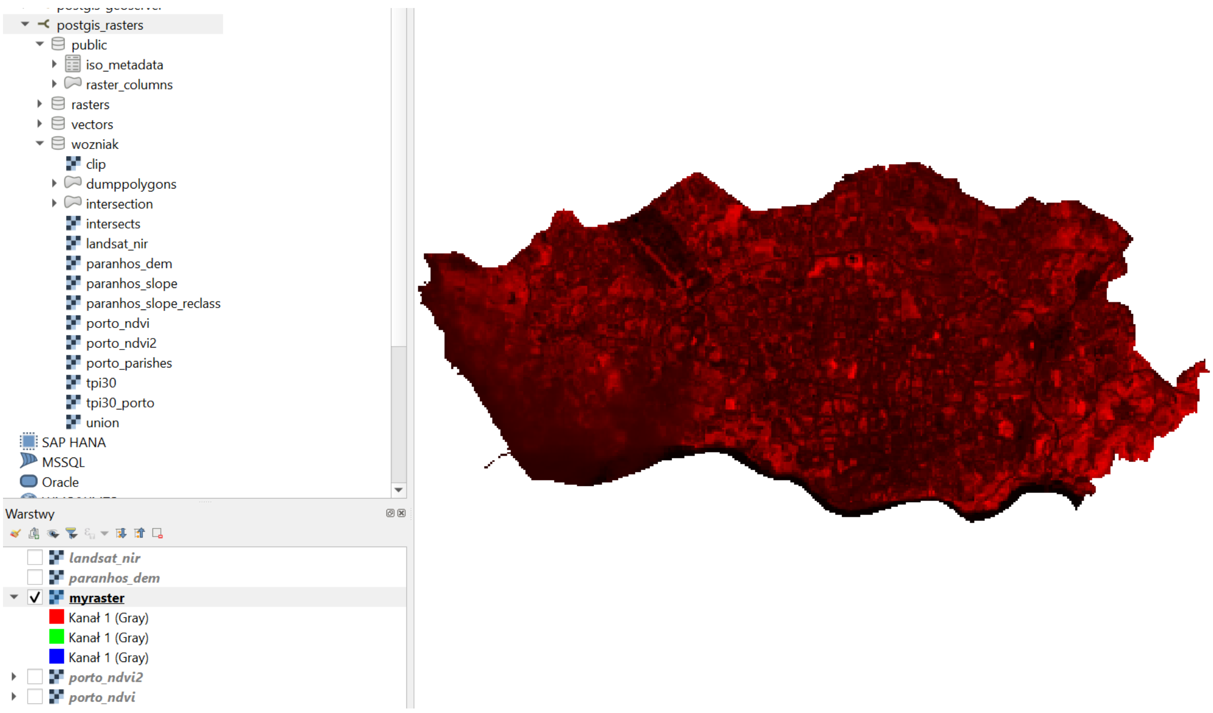
```
--Przykład 2 - ST_AsGDALRaster

SELECT ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE', 'PREDICTOR=2', 'PZLEVEL=9'])
FROM wozniak.porto_ndvi;


--Przykład 3 - Zapisywanie danych na dysku za pomocą dużego obiektu (large object, lo)

CREATE TABLE tmp_out AS
SELECT lo_from_bytea(0,
```

```
   st_asgdalraster
   bytea
1  [binary data]
```



```
--Przykład 3 - Zapisywanie danych na dysku za pomocą dużego obiektu (large object, lo)

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 wozniak.porto_ndvi;

SELECT lo_export(loid, 'E:\BDP\myraster.tiff') --> Save the file in a place where the user postgres
FROM tmp_out;

SELECT lo_unlink(loid)
FROM tmp_out; --> Delete the large object.



--Przykład 4 - Użycie Gdal

gdal_translate -co COMPRESS=DEFLATE -co PREDICTOR=2 -co ZLEVEL=9 PG:"host=localhost port=5432 dbname=
```

```
   lo_unlink
   integer
1          1
```


postgis_rasters

public

iso_metadata

raster_columns

rasters

vectors

wozniak

clip

dumppolygons

intersection

intersects

landsat_nir

paranhos_dem

paranhos_slope

paranhos_slope_reclass

porto_ndvi

porto_ndvi2

porto_parishes

tpi30

tpi30_porto

union

SAP HANA

MSSQL

Oracle

Warstwy

□ *landsat_nir*

□ *paranhos_dem*

☑ **myraster**

🟥 Kanał 1 (Gray)

🟩 Kanał 1 (Gray)

🟦 Kanał 1 (Gray)

□ *porto_ndvi2*

□ *porto_ndvi*