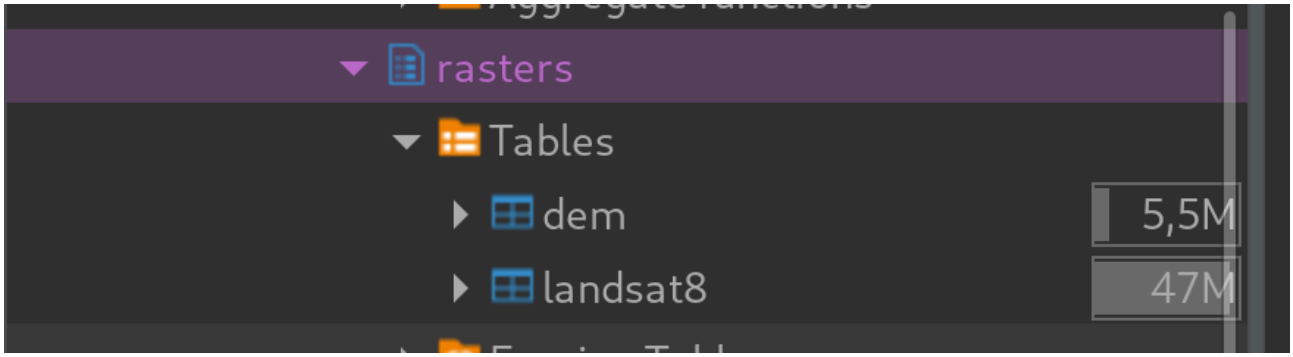
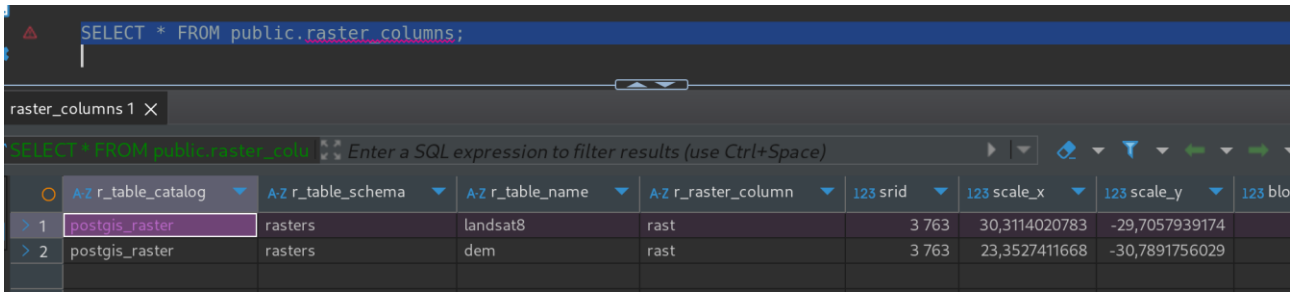
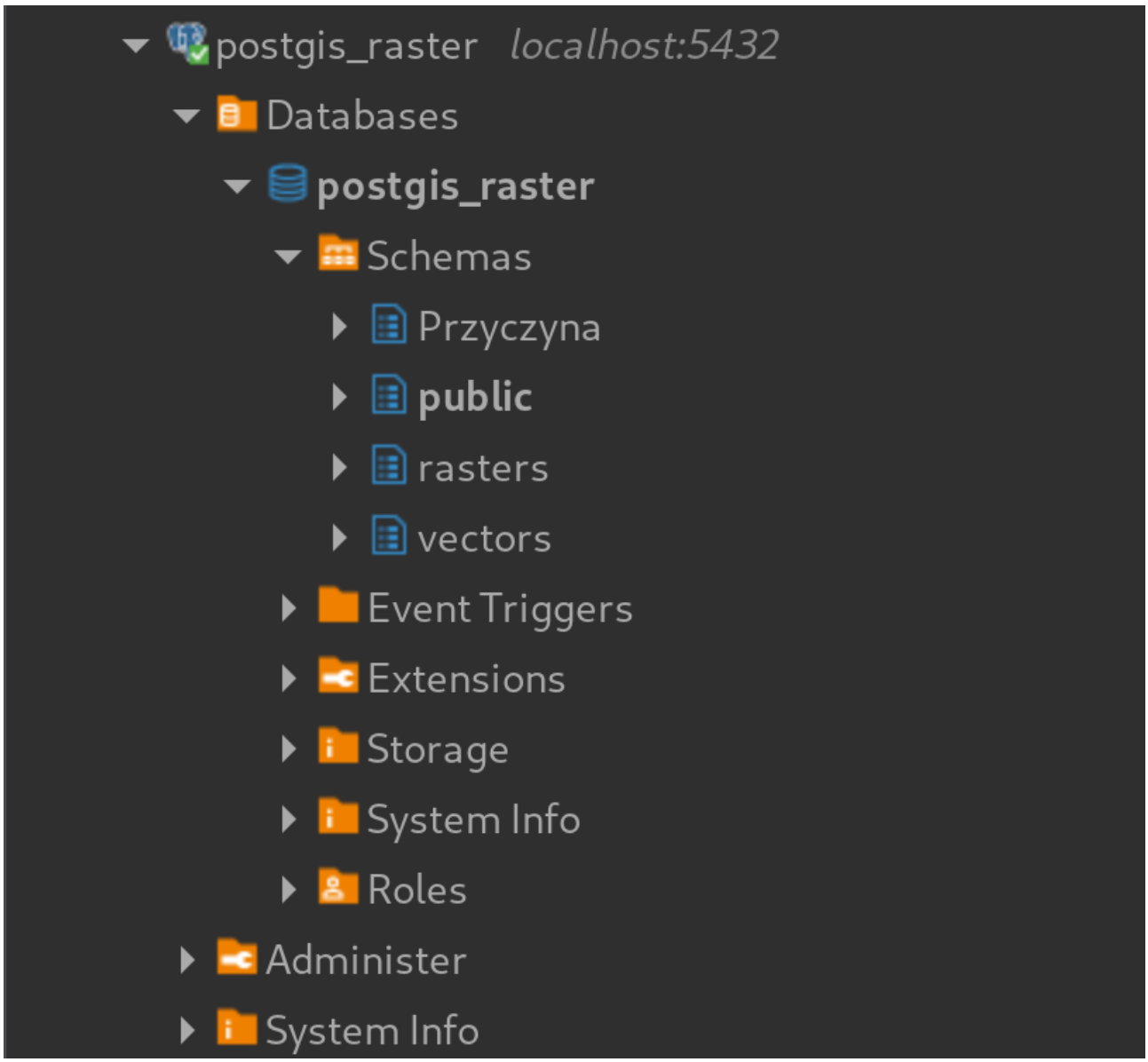
Ładowanie danych rastrowych




```
postgis_raster=# \dt rasters.*
              Lista relacji
 Schemat |  Nazwa   |  Typ   | Właściciel
---------+----------+--------+------------
 rasters | dem      | tabela | postgres
 rasters | landsat8 | tabela | postgres
(2 wiersze)
```





Tworzenie rastrów z istniejących rastrów i interakcja z wektorami

2. utworzenie indeksu przestrzennego:

```
CREATE TABLE "Przyczyna".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 "Przyczyna".intersects
add column rid SERIAL PRIMARY KEY;
```

Statistics 1

Name	Value
Updated Rows	0
Query	alter table "Przyczyna".intersects add column rid SERIAL PRIMARY KEY
Start time	Mon Nov 25 17:56:29 CET 2024
Finish time	Mon Nov 25 17:56:29 CET 2024

3. dodanie raster constraints:

```
alter table "Przyczyna".intersects
add column rid SERIAL PRIMARY KEY;

-- schema::name table_name::name raster_column::name
SELECT AddRasterConstraints('Przyczyna'::name,
'intersects'::name,'rast'::name);
```

Output

```
Enter a part of a message to search for here
Adding number of bands constraint
Adding pixel type constraint
Adding nodata value constraint
Adding out-of-database constraint
Adding maximum extent constraint
```

Results 1

```
SELECT AddRasterConstraints('Pr
```

Enter a SQL expression to filter results (use Ctrl+Space)

	addrasterconstraints
1	[v]

Obcinanie rastra na podstawie wektora.

```
CREATE TABLE "Przyczyna".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';
```

Statistics 1

Name	Value
Updated Rows	25
Query	CREATE TABLE "Przyczyna".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'
Start time	Mon Nov 25 17:59:08 CET 2024
Finish time	Mon Nov 25 17:59:09 CET 2024

Połączenie wielu kafelków w jeden raster.

```
CREATE TABLE "Przyczyna".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);
```

Statistics 1

Name	Value
Updated Rows	1
Query	CREATE TABLE "Przyczyna".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)
Start time	Mon Nov 25 17:59:59 CET 2024
Finish time	Mon Nov 25 17:59:59 CET 2024

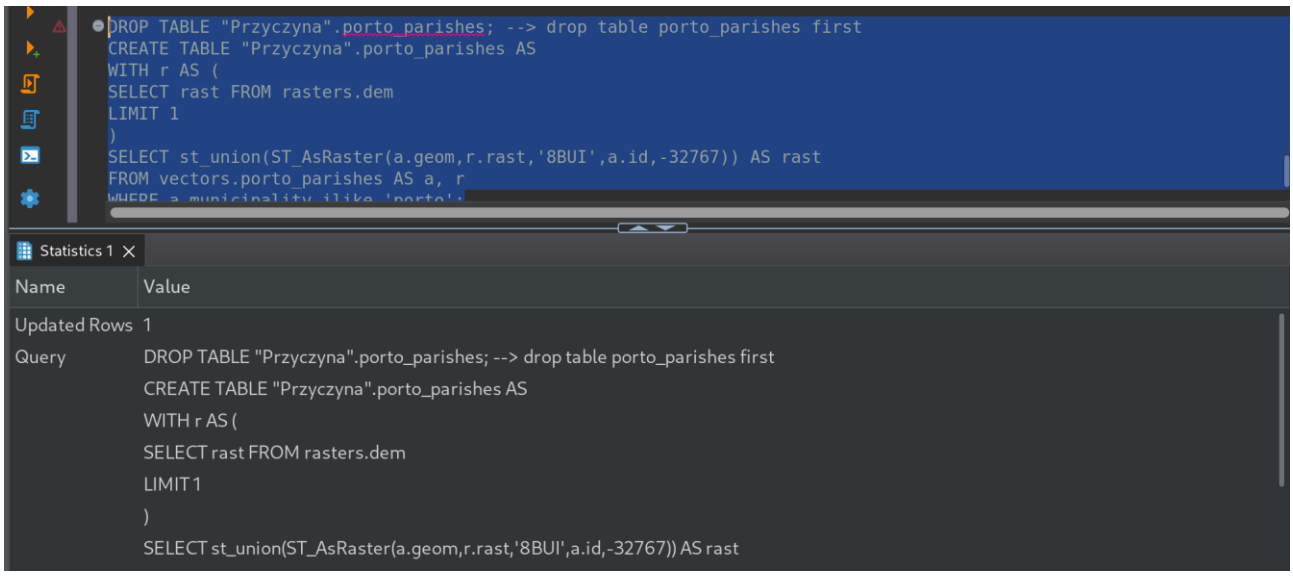
Przykład pokazuje użycie funkcji ST_AsRaster w celu rastrowania tabeli z parafiami o takiej samej charakterystyce przestrzennej tj.: wielkość piksela, zakresy itp.

```
CREATE TABLE "Przyczyna".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';
```

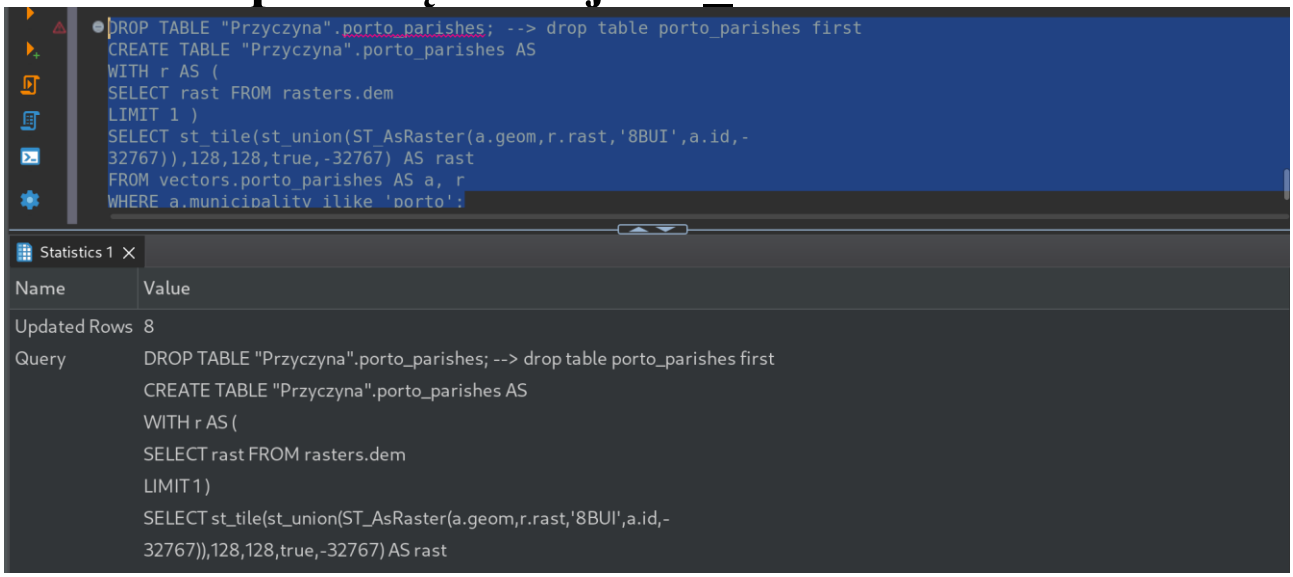
Statistics 1

Name	Value
Updated Rows	7
Query	CREATE TABLE "Przyczyna".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

Drugi przykład łączy rekordy z poprzedniego przykładu przy użyciu funkcji ST_UNION w pojedynczy raster.



Po uzyskaniu pojedynczego rastra można generować kafelki za pomocą funkcji ST_Tile.



Przykład 1 – ST_Intersection

```
create table "Przyczyna".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);
```

Statistics 1

Name	Value
Updated Rows	6649
Query	create table "Przyczyna".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)
Start time	Mon Nov 25 18:04:51 CET 2024

Przykład 2 – ST_DumpAsPolygons

```
CREATE TABLE "Przyczyna".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);
```

Statistics 1

Name	Value
Updated Rows	6442
Query	CREATE TABLE "Przyczyna".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)
Start time	Mon Nov 25 18:05:35 CET 2024
Finish time	Mon Nov 25 18:05:35 CET 2024

Scripts Scripts

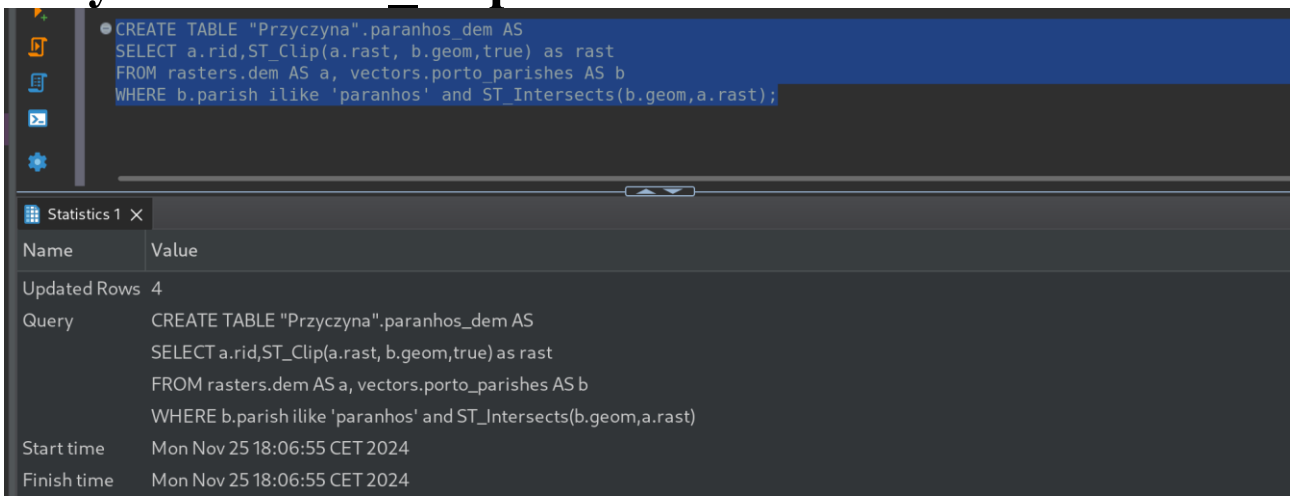
Przykład 1 – ST_Band

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

Statistics 1

Name	Value
Updated Rows	630
Query	CREATE TABLE "Przyczyna".landsat_nir AS SELECT rid, ST_Band(rast,4) AS rast FROM rasters.landsat8
Start time	Mon Nov 25 18:06:22 CET 2024
Finish time	Mon Nov 25 18:06:22 CET 2024

Przykład 2 – ST_Clip

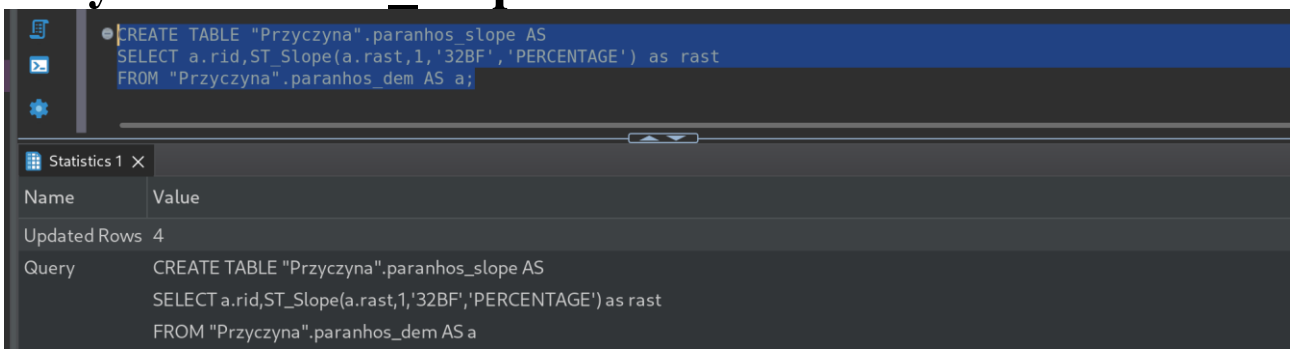


```
CREATE TABLE "Przyczyna".paranhos_dem AS
SELECT a.rid,ST_Clip(a.rast, b.geom,true) as rast
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Statistics 1

Name	Value
Updated Rows	4
Query	CREATE TABLE "Przyczyna".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)
Start time	Mon Nov 25 18:06:55 CET 2024
Finish time	Mon Nov 25 18:06:55 CET 2024

Przykład 3 – ST_Slope

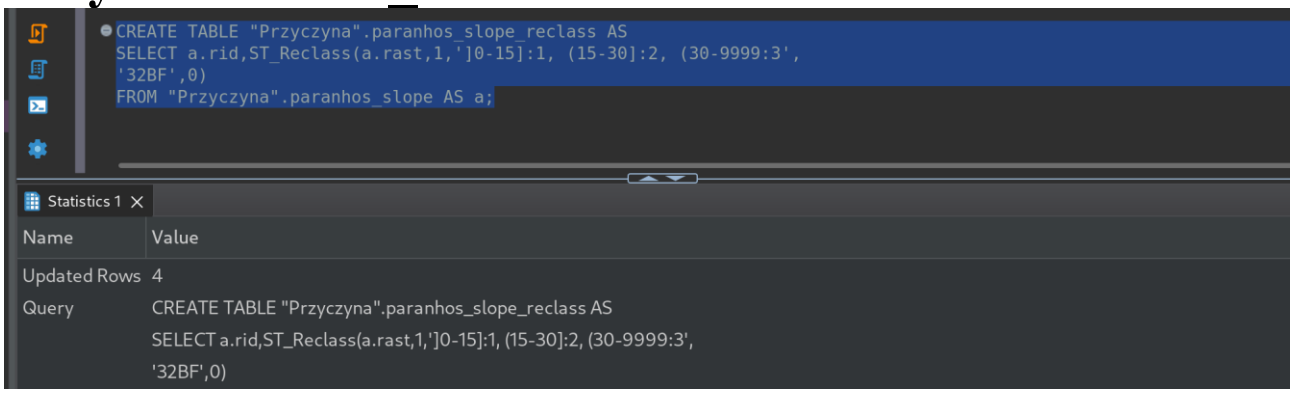


```
CREATE TABLE "Przyczyna".paranhos_slope AS
SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
FROM "Przyczyna".paranhos_dem AS a;
```

Statistics 1

Name	Value
Updated Rows	4
Query	CREATE TABLE "Przyczyna".paranhos_slope AS SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast FROM "Przyczyna".paranhos_dem AS a

Przykład 4 – ST_Reclass

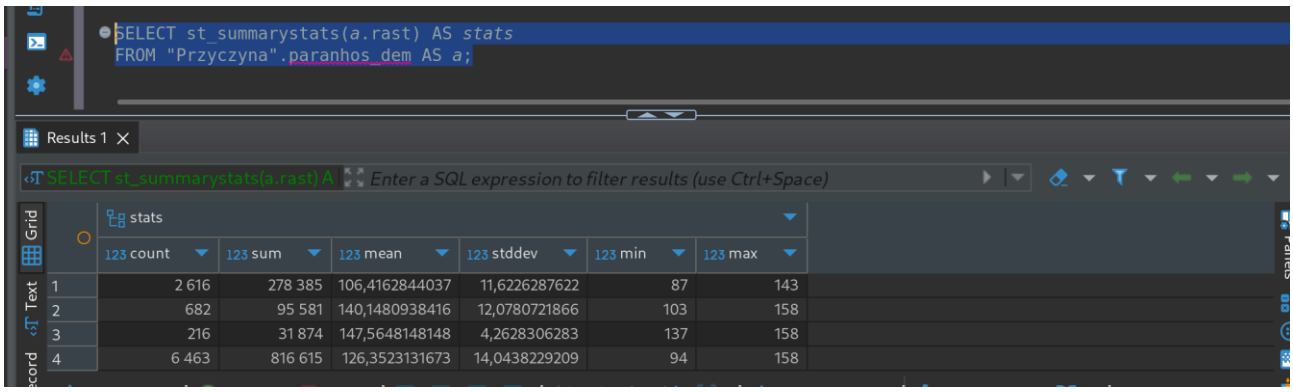


```
CREATE TABLE "Przyczyna".paranhos_slope_reclass AS
SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3',
'32BF',0)
FROM "Przyczyna".paranhos_slope AS a;
```

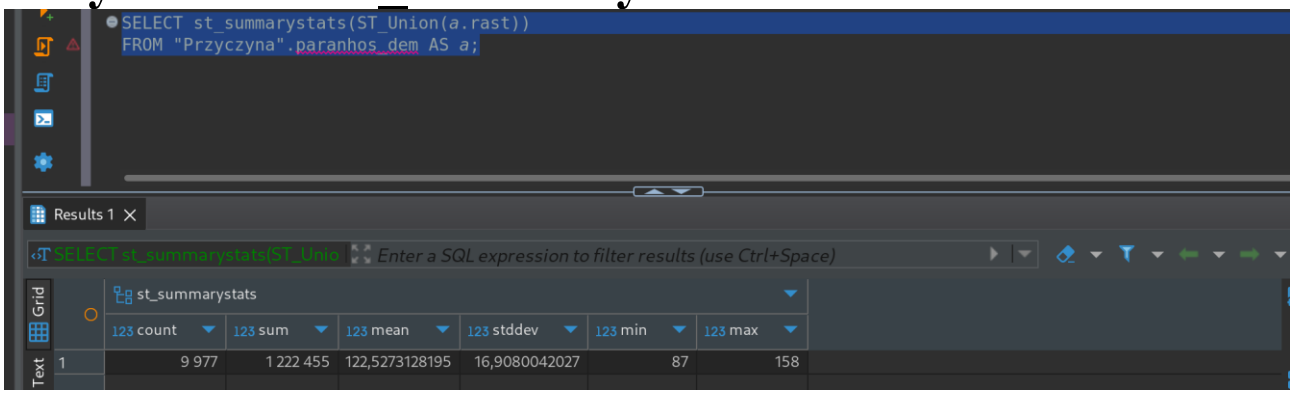
Statistics 1

Name	Value
Updated Rows	4
Query	CREATE TABLE "Przyczyna".paranhos_slope_reclass AS SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3', '32BF',0)

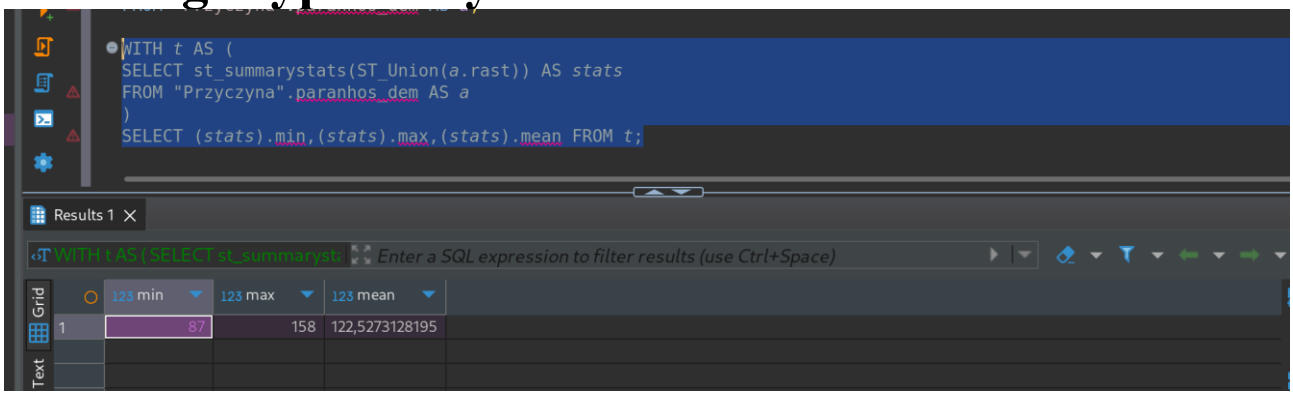
Przykład 5 – ST_SummaryStats



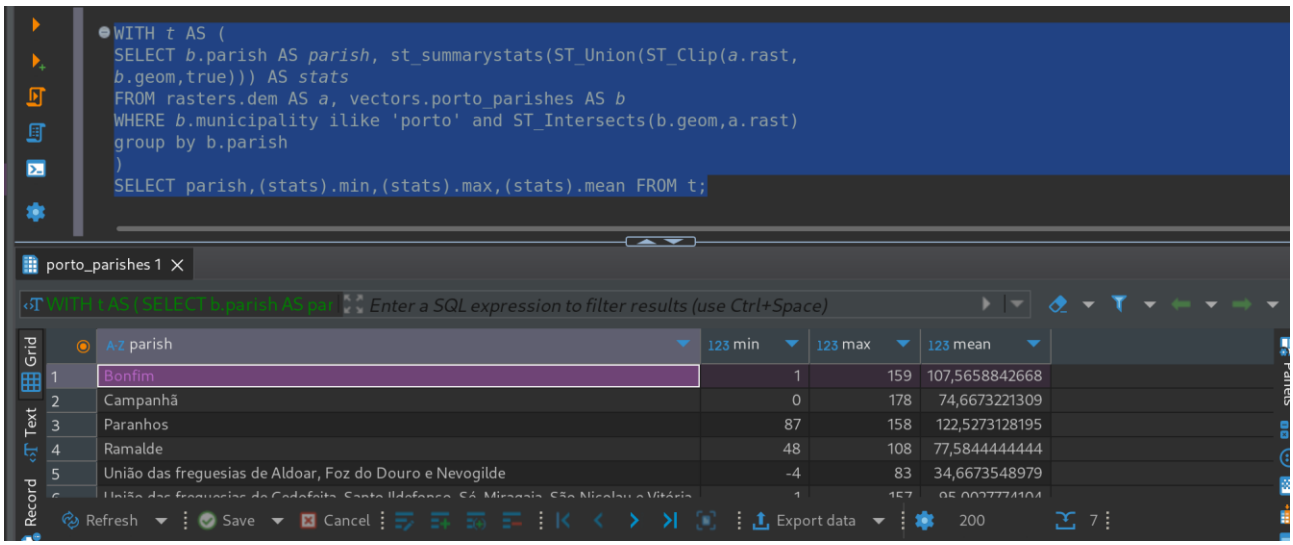
Przykład 6 - ST_SummaryStats oraz Union



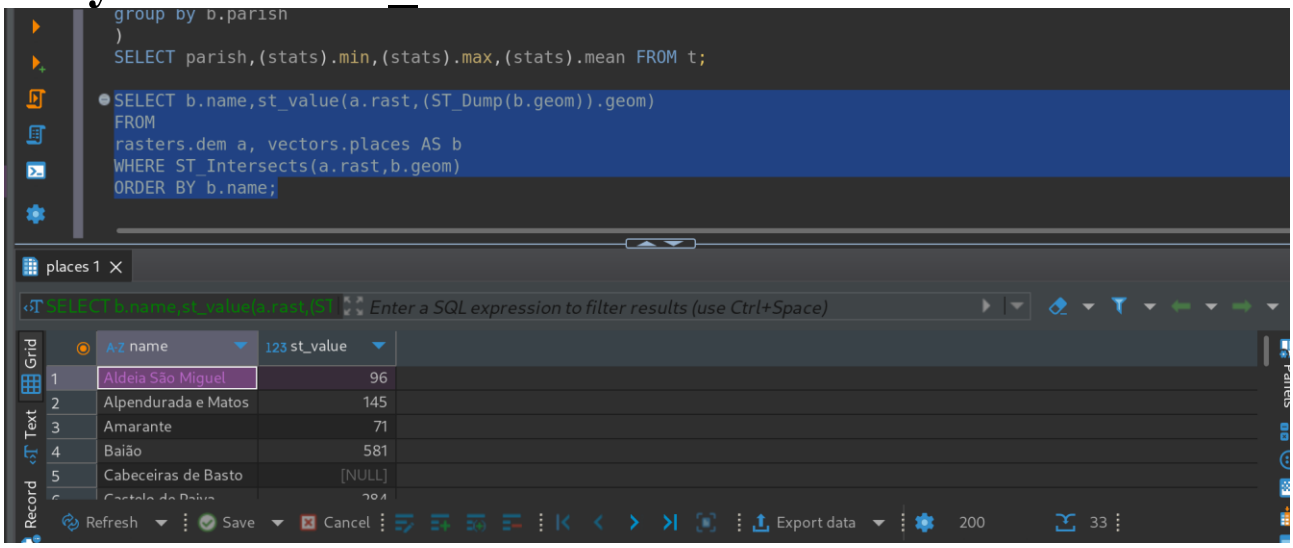
Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych



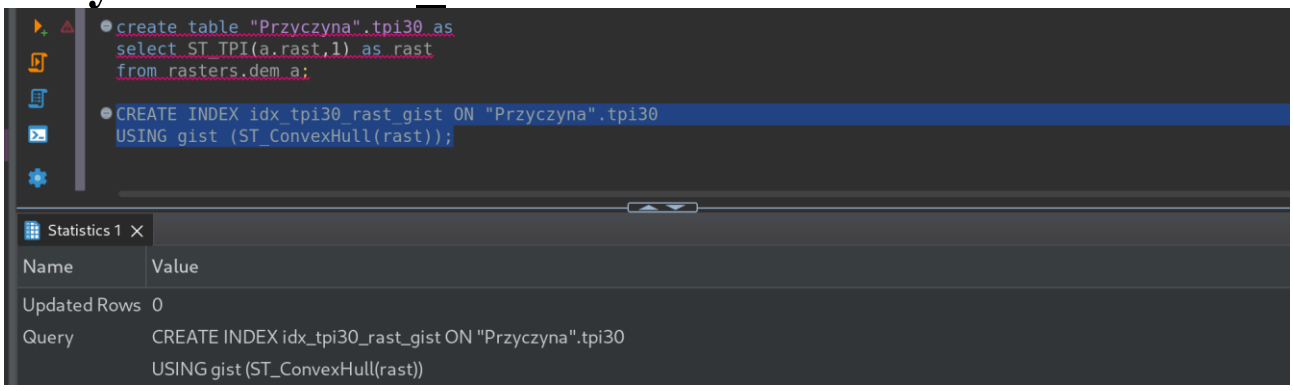
Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY



Przykład 9 – ST_Value



Przykład 10 – ST_TPI



```
EXPLAIN ANALYZE
SELECT
  ST_SummaryStats(rast)
FROM
  rasters.dem;
```

Results 1

EXPLAIN ANALYZE SELECT ST_Su | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	QUERY PLAN
1	Seq Scan on dem (cost=0.00..162.14 rows=589 width=32) (actual time=0.548..298.342 rows=589 loops=1)
2	Planning Time: 0.136 ms
3	Execution Time: 298.442 ms

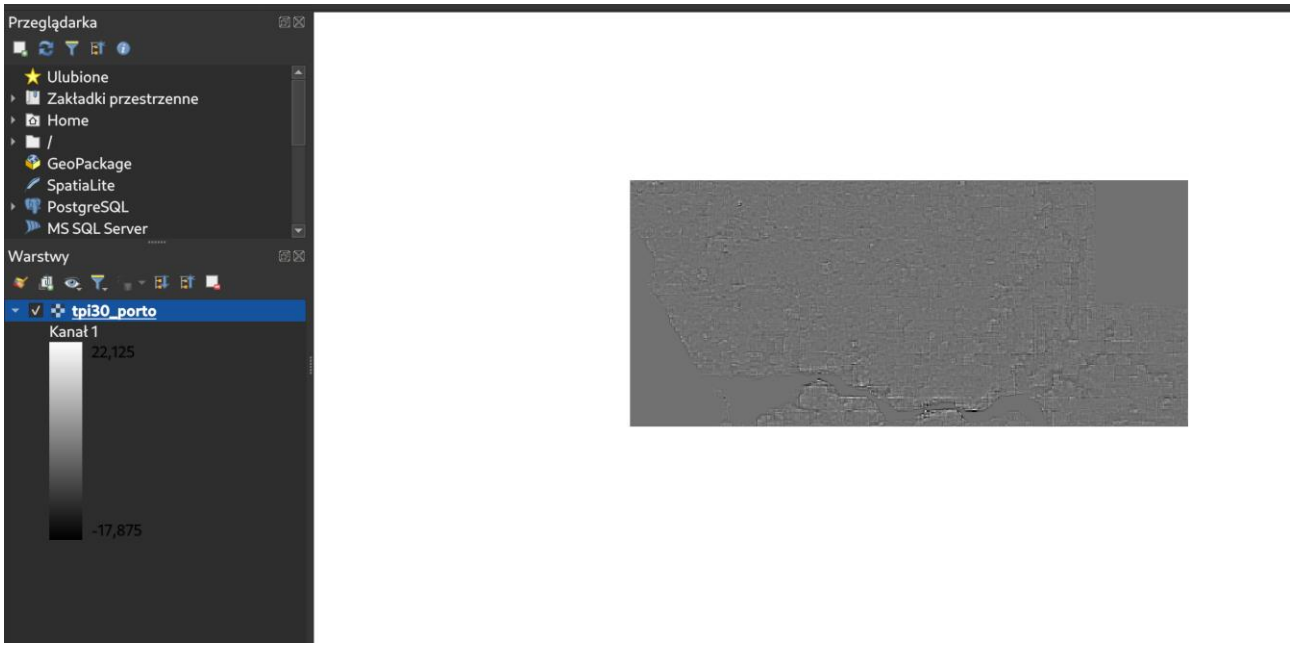
```
EXPLAIN ANALYZE
SELECT
  ST_SummaryStats(clipped_raster)
FROM
  (SELECT ST_Clip(a.rast, b.geom) AS clipped_raster
   FROM rasters.dem a
   JOIN vectors.porto_parishes b
   ON ST_Intersects(a.rast, b.geom)
   WHERE b.municipality ILIKE 'porto') AS clipped;
```

Results 1

EXPLAIN ANALYZE SELECT ST_Su | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text

	QUERY PLAN
7	Filter: _st_intersects(b.geom, rast, NULL::integer)
8	Rows Removed by Filter: 0
9	Planning Time: 0.370 ms
10	Execution Time: 37.421 ms



Przykład 1 - Wyrażenie Algebry Map

```
CREATE TABLE "Przyczyna".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,
```

Statistics 1

Name	Value
Updated Rows	29
Query	CREATE TABLE "Przyczyna".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

Indeks przestrzenny

Properties | ER Diagram | postgis_raster | Databases | postgis_raster | Schemas | Przyczyna

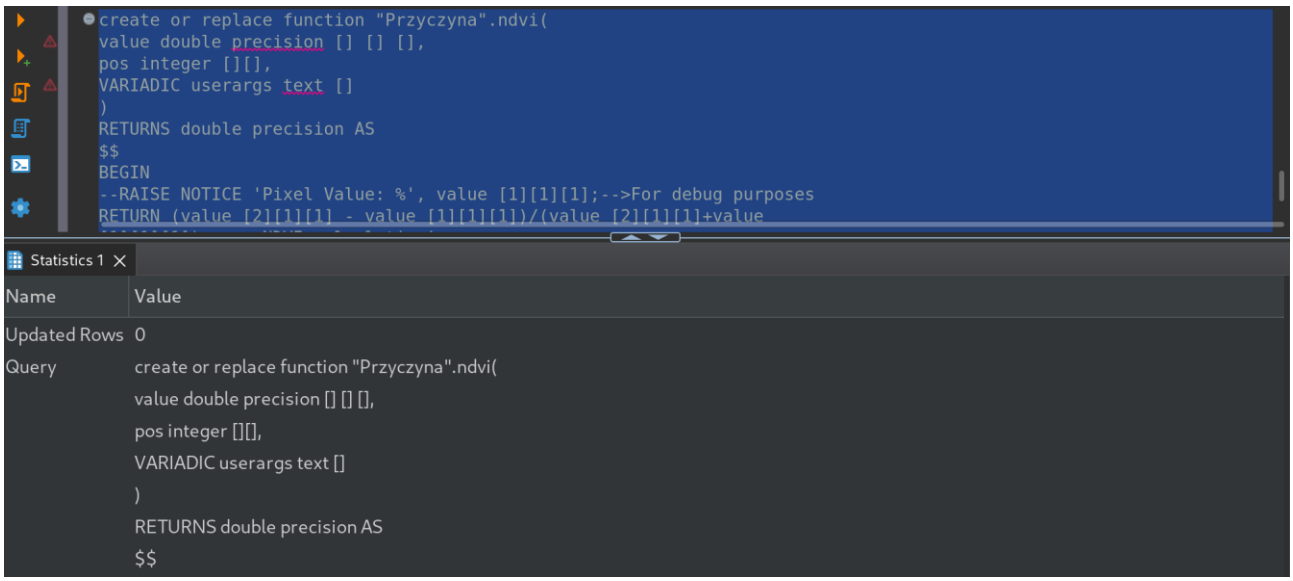
Name: Przyczyna
Comment:
Namespace ID: 29778
Owner: postgres

Tables
Foreign Tables
Views
Materialized Views
Indexes
Functions
Sequences
Data types
Aggregate functions
Permissions
Source

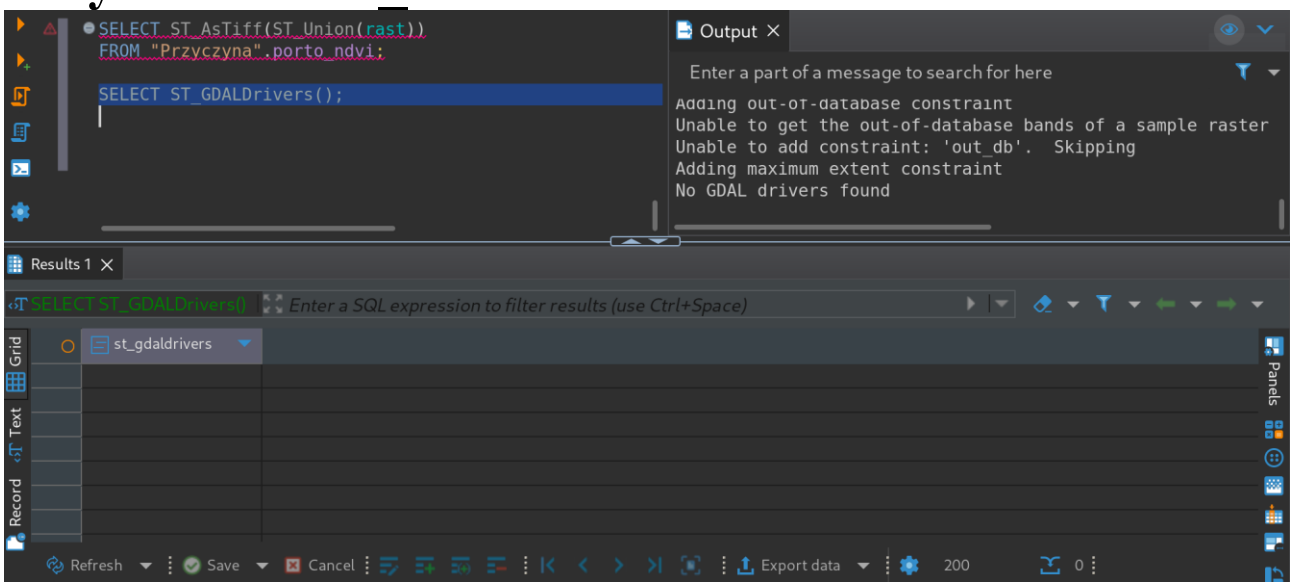
Table Name	Object ID	Owner	Tablespace	Row Count Estimate	Has Row-Level Security	Partitions	Partitior
clip	34 151	postgres	pg_default	-1	[ ]	[ ]	
dumppolygon:	34 199	postgres	pg_default	6 442	[ ]	[ ]	
intersection	34 191	postgres	pg_default	6 649	[ ]	[ ]	
intersects	34 075	postgres	pg_default	25	[ ]	[ ]	
landsat_nir	34 204	postgres	pg_default	630	[ ]	[ ]	
paranhos_den	34 839	postgres	pg_default	-1	[ ]	[ ]	
paranhos_slop	34 846	postgres	pg_default	-1	[ ]	[ ]	
paranhos_slop	34 854	postgres	pg_default	-1	[ ]	[ ]	
porto_ndvi	35 375	postgres	pg_default	29	[ ]	[ ]	
porto_parishe:	34 186	postgres	pg_default	-1	[ ]	[ ]	
tpi30	34 859	postgres	pg_default	589	[ ]	[ ]	
union	34 169	postgres	pg_default	-1	[ ]	[ ]	

Refresh | Save ... | Revert

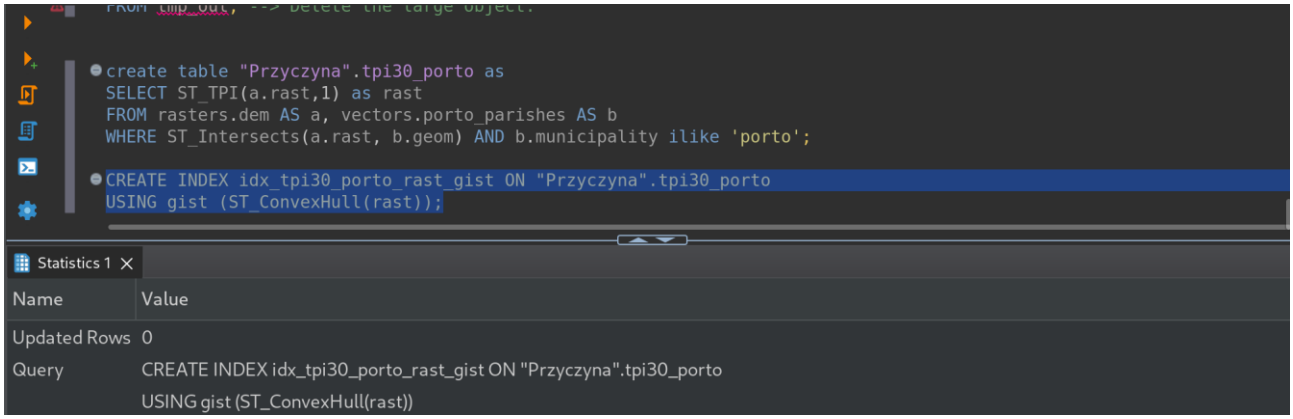
Przykład 2 – Funkcja zwrotna



Przykład 2 - ST_AsGDALRaster



Rozwiązanie problemu z poprzedniej części


FROM tmp_out; --> Delete the large object.
create table "Przyczyna".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 "Przyczyna".tpi30_porto
USING gist (ST_ConvexHull(rast));
Statistics 1
Name
Value
Updated Rows
0
Query
CREATE INDEX idx_tpi30_porto_rast_gist ON "Przyczyna".tpi30_porto
USING gist (ST_ConvexHull(rast))