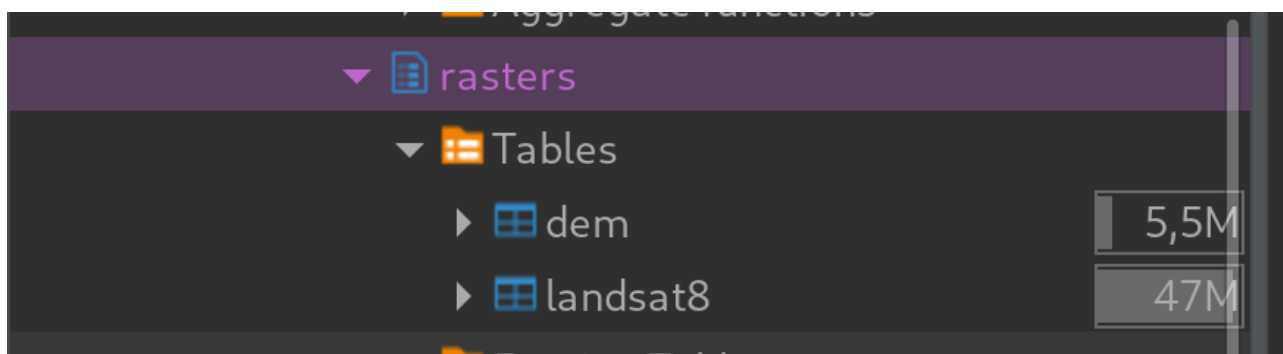


Ładowanie danych rastrowych

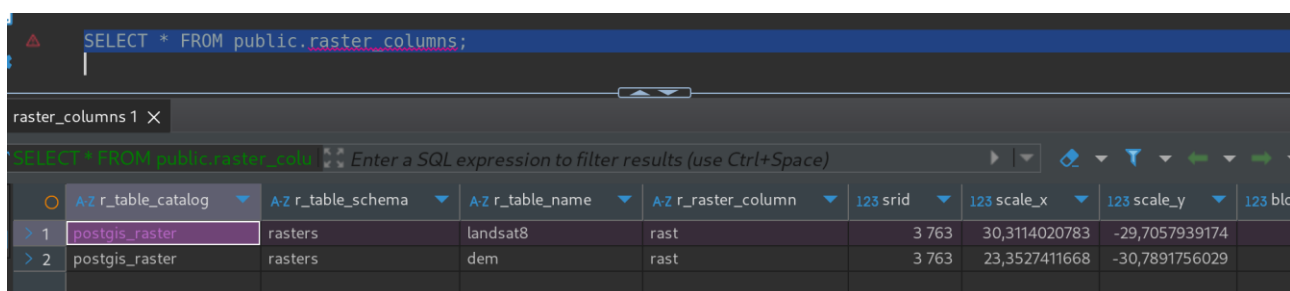


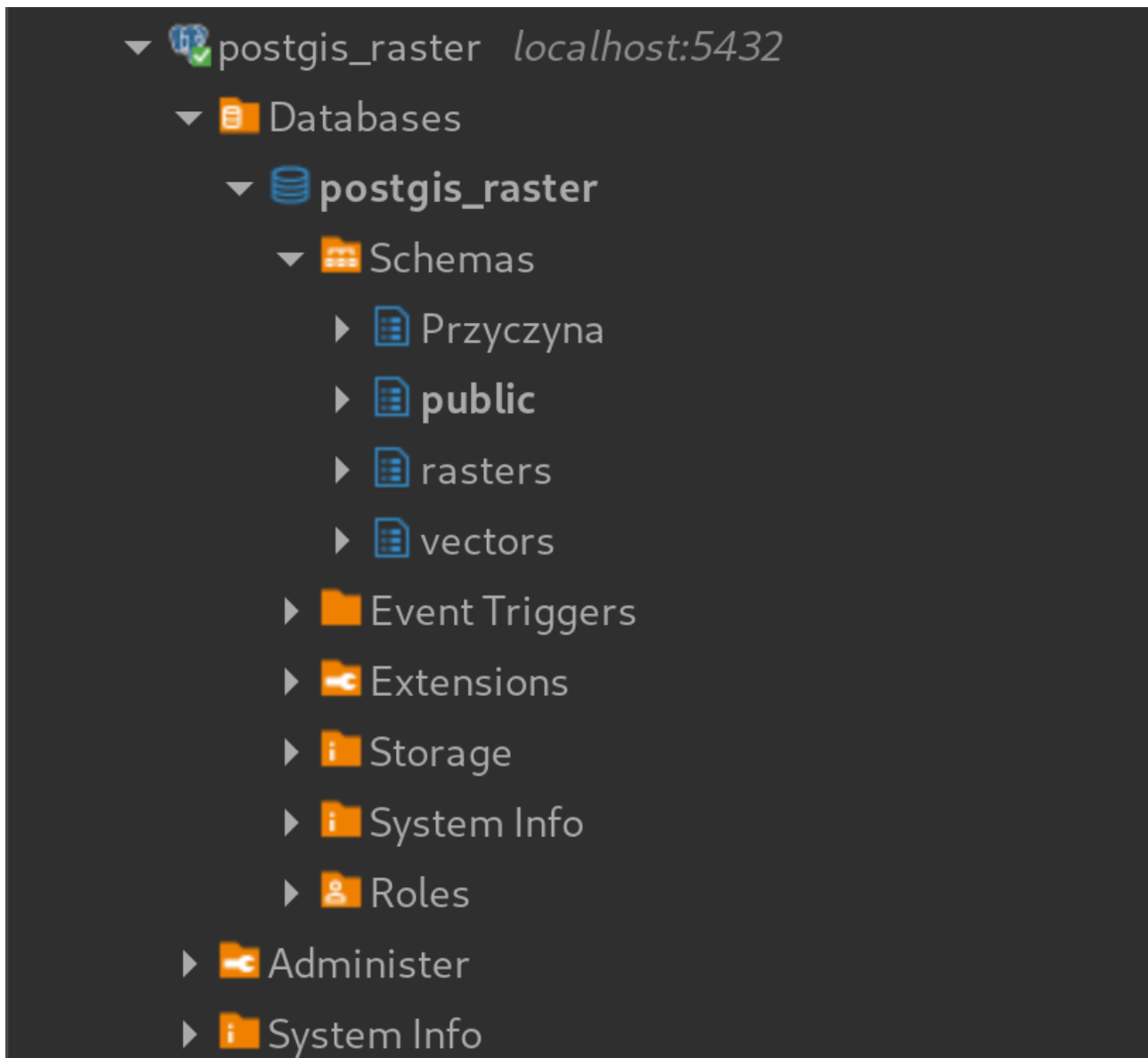
```
postgis_raster=# \dt rasters.*
```

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;
```

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 X

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 X

SELECT AddRasterConstraints('Pr: Enter a SQL expression to filter results (use Ctrl+Space)

Grid

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';
```

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);
```

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';
```

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 WHERE a.municipality ilike 'porto';

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

```
● DROP TABLE "Przyczyna".porto_parishes; --> drop table porto_parishes first
CREATE TABLE "Przyczyna".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'
```

Name	Value
Updated Rows	1
Query	DROP TABLE "Przyczyna".porto_parishes; --> drop table porto_parishes first CREATE TABLE "Przyczyna".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

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

```
● DROP TABLE "Przyczyna".porto_parishes; --> drop table porto_parishes first
CREATE TABLE "Przyczyna".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'
```

Name	Value
Updated Rows	8
Query	DROP TABLE "Przyczyna".porto_parishes; --> drop table porto_parishes first CREATE TABLE "Przyczyna".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

Przykład 1 – ST_Intersection

The screenshot shows a SQL query editor with the following 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);
```

Below the query editor is a 'Statistics 1' window with the following data:

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

The screenshot shows a SQL query editor with the following 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);
```

Below the query editor is a 'Statistics 1' window with the following data:

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

Przykład 1 – ST_Band

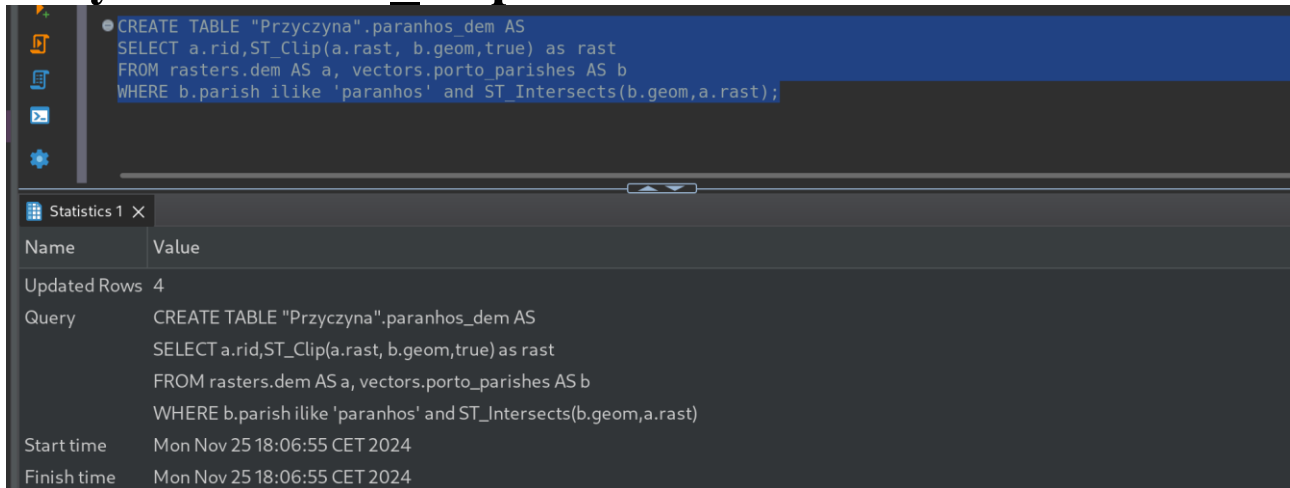
The screenshot shows a SQL query editor with the following query:

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

Below the query editor is a 'Statistics 1' window with the following data:

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



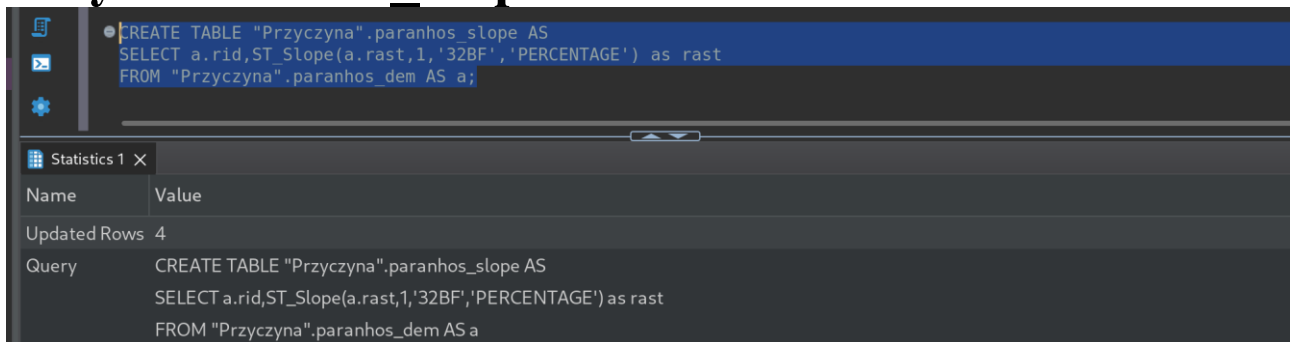
The screenshot shows a GIS software interface with a SQL editor and a statistics panel. The SQL editor contains the following query:

```
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);
```

The statistics panel, titled "Statistics 1 X", displays the following information:

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



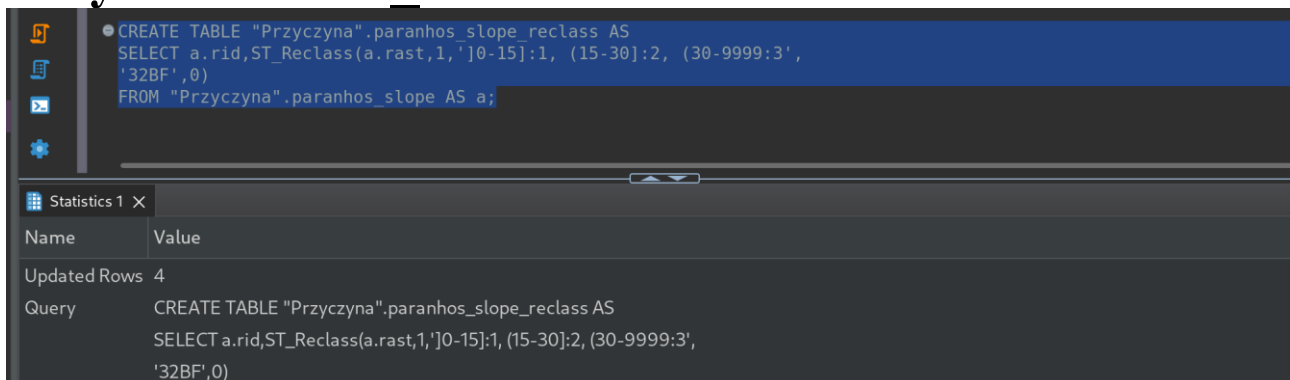
The screenshot shows a GIS software interface with a SQL editor and a statistics panel. The SQL editor contains the following query:

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

The statistics panel, titled "Statistics 1 X", displays the following information:

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



The screenshot shows a GIS software interface with a SQL editor and a statistics panel. The SQL editor contains the following 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)
FROM "Przyczyna".paranhos_slope AS a;
```

The statistics panel, titled "Statistics 1 X", displays the following information:

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

SQL Query:

```
SELECT st_summarystats(a.rast) AS stats
FROM "Przyczyna".paranhos_dem AS a;
```

Results 1 X

SQL Expression to filter results (use Ctrl+Space):

```
SELECT st_summarystats(rast) AS stats
```

	stats
	123 count 123 sum 123 mean 123 stddev 123 min 123 max
1	2 616 278 385 106,4162844037 11,6226287622 87 143
2	682 95 581 140,1480938416 12,0780721866 103 158
3	216 31 874 147,5648148148 4,2628306283 137 158
4	6 463 816 615 126,3523131673 14,0438229209 94 158

Przykład 6 - ST_SummaryStats oraz Union

SQL Query:

```
SELECT st_summarystats(ST_Union(a.rast))
FROM "Przyczyna".paranhos_dem AS a;
```

Results 1 X

SQL Expression to filter results (use Ctrl+Space):

```
SELECT st_summarystats(ST_Union(rast))
```

	st_summarystats
	123 count 123 sum 123 mean 123 stddev 123 min 123 max
1	9 977 1 222 455 122,5273128195 16,9080042027 87 158

Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych

SQL Query:

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

Results 1 X

SQL Expression to filter results (use Ctrl+Space):

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

	123 min	123 max	123 mean
1	87	158	122,5273128195

Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY

SQL Query:

```
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;
```

Results (porto_parishes 1 X):

A-Z parish	123 min	123 max	123 mean
1 Bonfim	1	159	107,5658842668
2 Campanhã	0	178	74,6673221309
3 Paranhos	87	158	122,5273128195
4 Ramalde	48	108	77,5844444444
5 União das freguesias de Aldoar, Foz do Douro e Nevogilde	-4	83	34,6673548979
6 União das freguesias de Cadafaz, Santa Ildefonso, São Mamede, São Nicolau e Vitória	1	157	95,007774104

Przykład 9 – ST_Value

SQL Query:

```
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;
```

Results (places 1 X):

A-Z name	123 st_value
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	104

Przykład 10 – ST_TPI

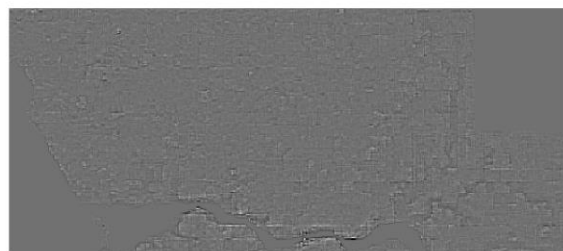
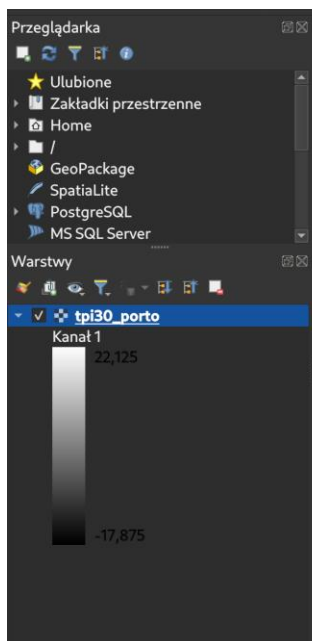
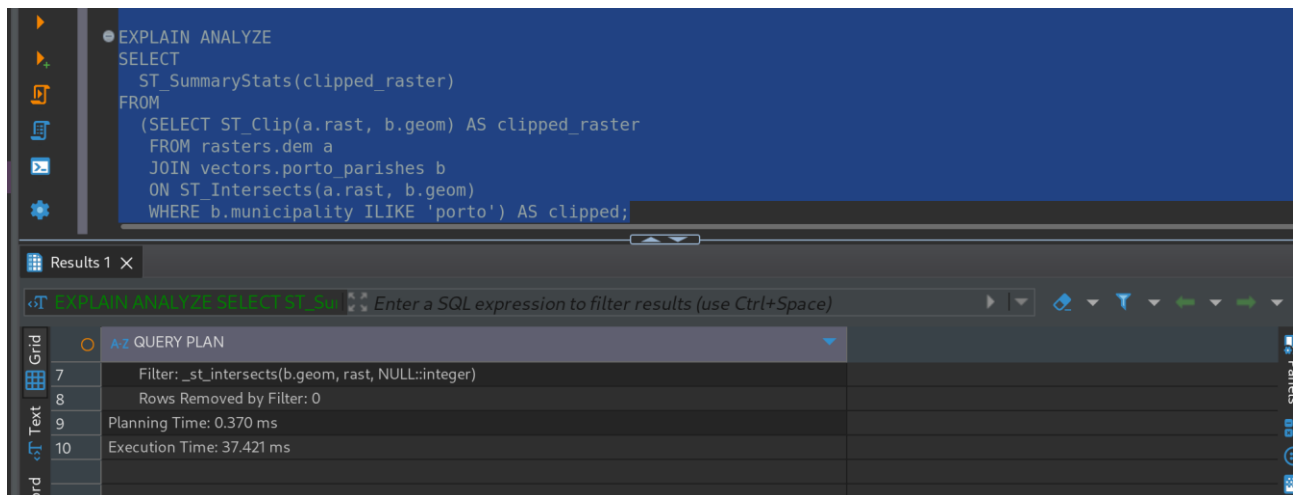
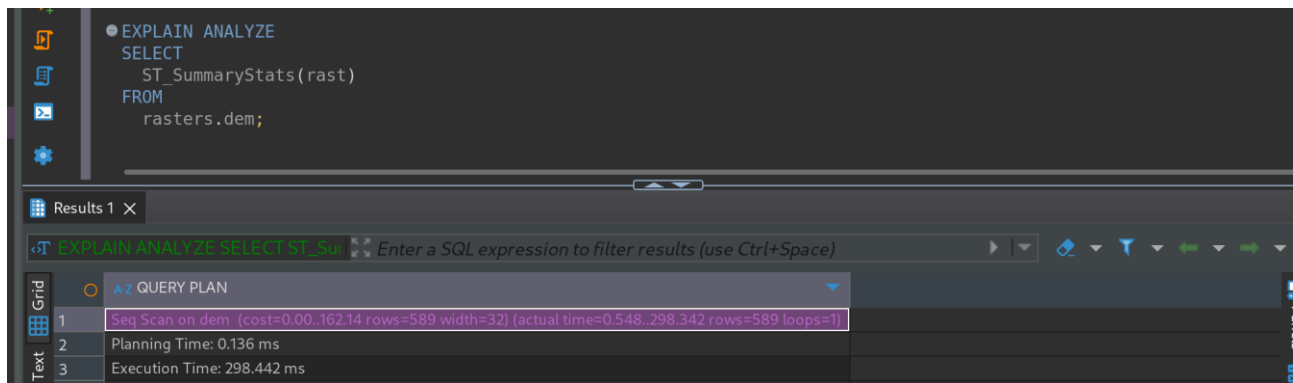
SQL Query:

```
create table "Przyczyna".tpi30 as
select ST_TPI(a.rast,1) as rast
from rasters.dem a;

CREATE INDEX idx_tpi30_rast_gist ON "Przyczyna".tpi30
USING gist (ST_ConvexHull(rast));
```

Statistics 1 X:

Name	Value
Updated Rows	0
Query	CREATE INDEX idx_tpi30_rast_gist ON "Przyczyna".tpi30 USING gist (ST_ConvexHull(rast))



Przykład 1 - Wyrażenie Algebra 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,
rast * (1/rast) null, rast * (1/rast) null) / (rast) null

```

Statistics 1 X

Name	Value
Updated Rows	29
Query	<pre> 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 </pre>

Indeks przestrzenny

Properties ER Diagram postgis_raster Databases postgis_raster Schemas Przyczyna

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

	Table Name	Object ID	Owner	Tablespace	Row Count Estimate	Has Row-Level Security	Partitions	Partition
Foreign Tables	clip	34 151	postgres	pg_default	-1	[]	[]	
Views	dumppolygon:	34 199	postgres	pg_default	6 442	[]	[]	
Materialized Views	intersection	34 191	postgres	pg_default	6 649	[]	[]	
Indexes	intersects	34 075	postgres	pg_default	25	[]	[]	
Functions	landsat_nir	34 204	postgres	pg_default	630	[]	[]	
Sequences	paranhos_derr	34 839	postgres	pg_default	-1	[]	[]	
Data types	paranhos_slop	34 846	postgres	pg_default	-1	[]	[]	
Aggregate functions	paranhos_slop	34 854	postgres	pg_default	-1	[]	[]	
Permissions	porto_ndvi	35 375	postgres	pg_default	29	[]	[]	
Source	porto_parishes	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

```

create or replace function "Przyczyna".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

```

Name	Value
Updated Rows	0
Query	<pre> create or replace function "Przyczyna".ndvi(value double precision [] [] [], pos integer [], VARIADIC userargs text []) RETURNS double precision AS \$\$ </pre>

Przykład 2 - ST_AsGDALRaster

```

SELECT ST_AsTiff(ST_Union(rast))
FROM "Przyczyna".porto_ndvi;

SELECT ST_GDALDrivers();

```

Output X

```

Enter a part of a message to search for here
Adding out-of-database constraint
Unable to get the out-of-database bands of a sample raster
Unable to add constraint: 'out_db'. Skipping
Adding maximum extent constraint
No GDAL drivers found

```

Name	Value
Updated Rows	0
Query	<pre> SELECT ST_GDALDrivers(); </pre>

Rozwiązanie problemu z poprzedniej części

```

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';

```

Name	Value
Updated Rows	25
Query	<pre> 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' </pre>

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 X

Name	Value
Updated Rows	0
Query	CREATE INDEX idx_tpi30_porto_rast_gist ON "Przyczyna".tpi30_porto USING gist (ST_ConvexHull(rast))