

Tworzenie rastrow z istniejących rastrow i interakcja z wektorami

Przykład 1 - ST_Intersects

```
CREATE EXTENSION postgis;
CREATE EXTENSION postgis_raster;

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

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

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

results 1 ×

SELECT AddRasterConstraints('schema_Slazyk'::name, 'intersects'::name, 'rast'::name)

	addrasterconstraints	Value
1	[v]	true













































Przykład 2 - ST_Clip

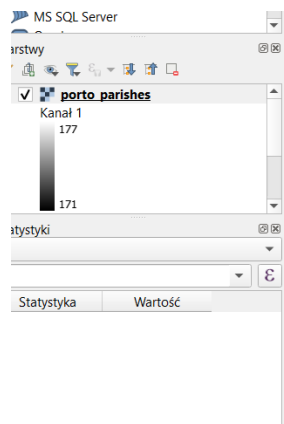
```
CREATE TABLE "schema_Slazyk".clip AS
SELECT ST_Clip(a.rast, b.geom, true), b.municipality
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE ST_Intersects(a.rast, b.geom) AND b.municipality like 'PORTO';
```

Statistics 1 ×

Name	Value
Updated Rows	29
Query	CREATE TABLE "schema_Slazyk".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'
Finish time	Sun Dec 04 11:47:38 CET 2022

Przykład 3 - ST_Union

	<pre>CREATE TABLE "schema_Slazyk".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);</pre>
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
	
<	



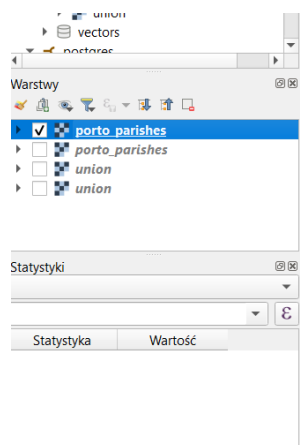
Przykład 2 - ST_Union

```

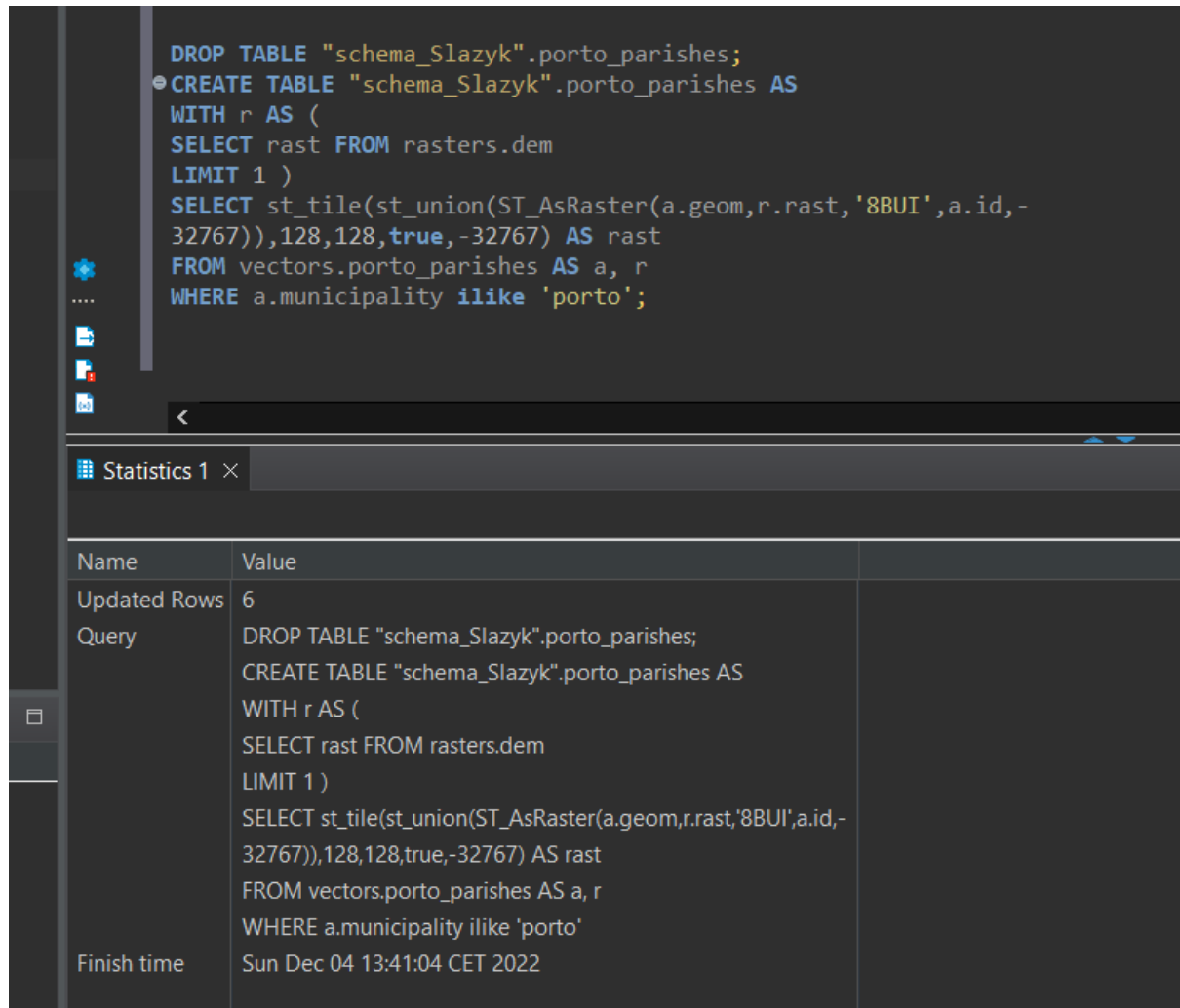
DROP TABLE "schema_Slazyk".porto_parishes;
CREATE TABLE "schema_Slazyk".porto_parishes AS
WITH r AS (
  SELECT rast FROM rasters.dem
  LIMIT 1
)
SELECT st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767)) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality ilike 'porto';

```

Name	Value
Updated Rows	1
Query	DROP TABLE "schema_Slazyk".porto_parishes; CREATE TABLE "schema_Slazyk".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'
Finish time	Sun Dec 04 13:37:31 CET 2022



Przykład 3 - ST_Tile



The screenshot shows a SQL IDE interface. The top pane contains a SQL query that creates a table named 'porto_parishes' in the 'schema_Slazyk' schema. The query uses a Common Table Expression (CTE) named 'r' to select a single raster from 'rasters.dem'. It then uses the 'ST_Tile' function to tile the union of the geometry from 'vectors.porto_parishes' and the raster 'r' into 128x128 pixel tiles, with a bounding box of approximately 32767 to -32767. The tiles are stored in the 'porto_parishes' table. The bottom pane shows the execution statistics for this query.

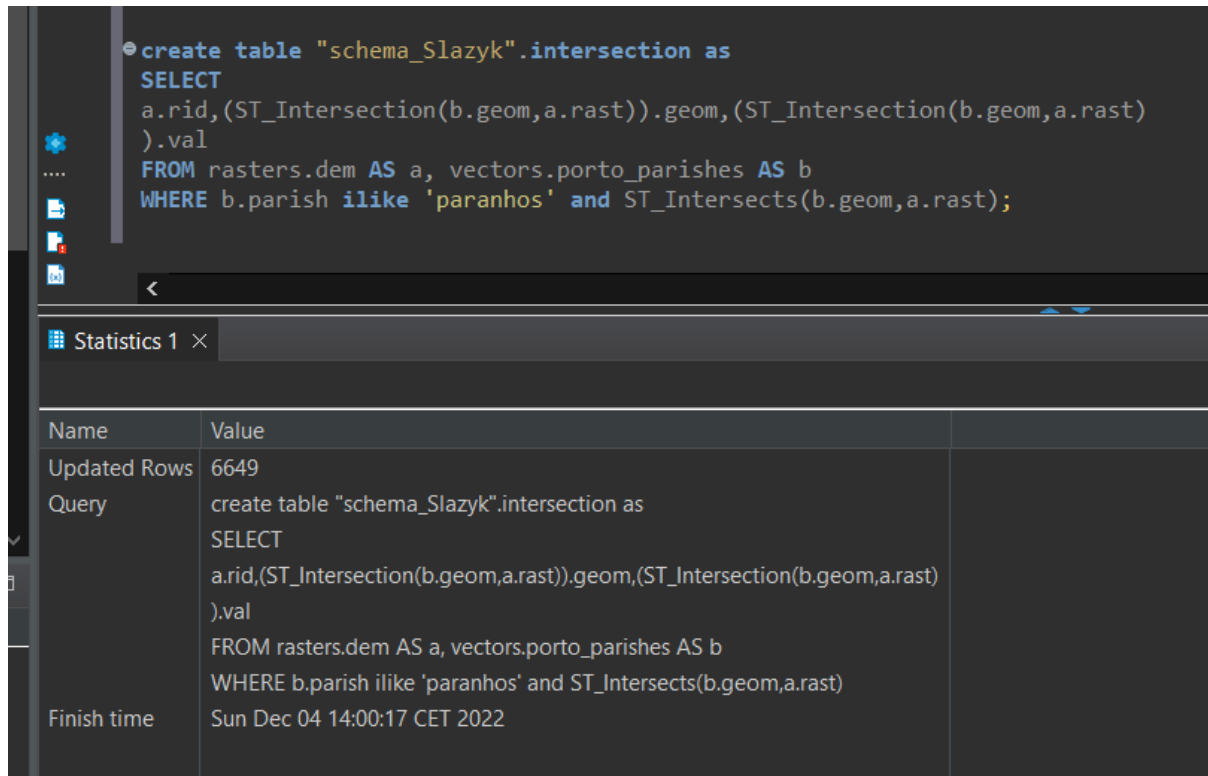
```
DROP TABLE "schema_Slazyk".porto_parishes;
CREATE TABLE "schema_Slazyk".porto_parishes AS
WITH r AS (
  SELECT rast FROM rasters.dem
  LIMIT 1 )
SELECT st_tile(st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-
32767)),128,128,true,-32767) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality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

Statistics 1 ×

Name	Value
Updated Rows	6
Query	DROP TABLE "schema_Slazyk".porto_parishes; CREATE TABLE "schema_Slazyk".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'
Finish time	Sun Dec 04 13:41:04 CET 2022

Konwertowanie rastrów na wektory (wektoryzowanie)

Przykład 1 - ST_Intersection



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

Name	Value
Updated Rows	6649
Query	create table "schema_Slazyk".intersection as SELECT a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast) .val FROM rasters.dem AS a, vectors.porto_parishes AS b WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast)
Finish time	Sun Dec 04 14:00:17 CET 2022

Przykład 2 - ST_DumpAsPolygons

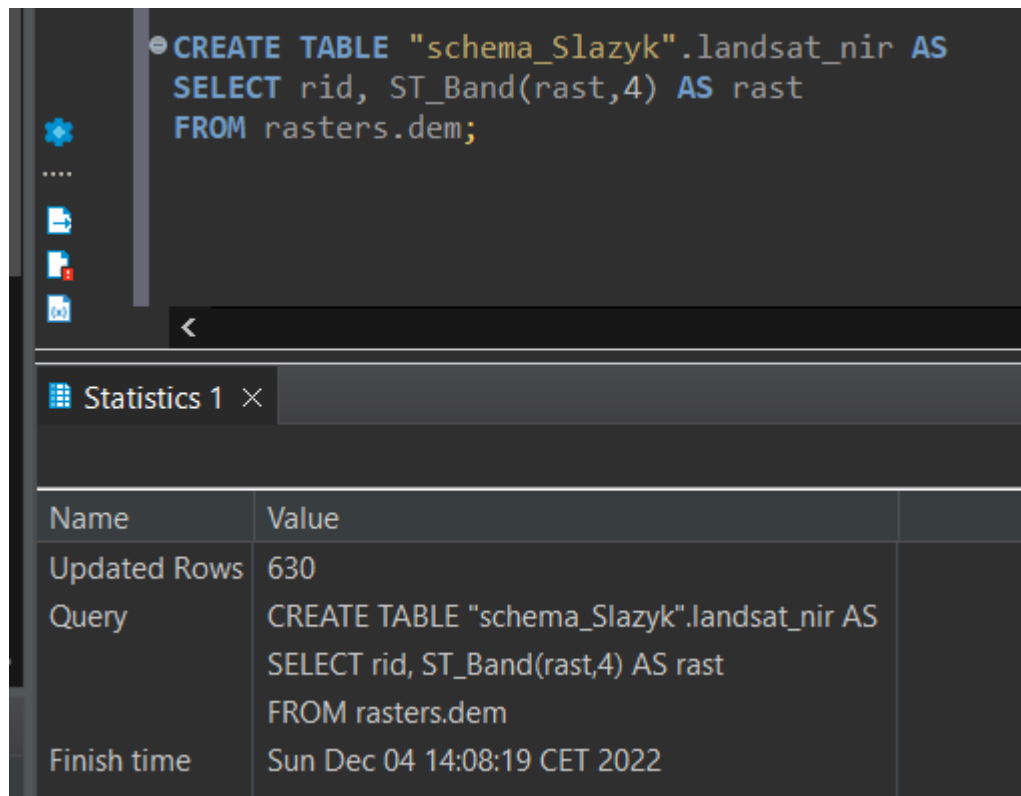


```
CREATE TABLE "schema_Slazyk".dumppolygons AS
SELECT
a.rid,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).geom,(ST_DumpAsPolygons(S
T_Clip(a.rast,b.geom))).val
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Name	Value
Updated Rows	6649
Query	create table "schema_Slazyk".intersection as SELECT a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast) .val FROM rasters.dem AS a, vectors.porto_parishes AS b WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast)
Finish time	Sun Dec 04 14:00:17 CET 2022

Analiza rastrów

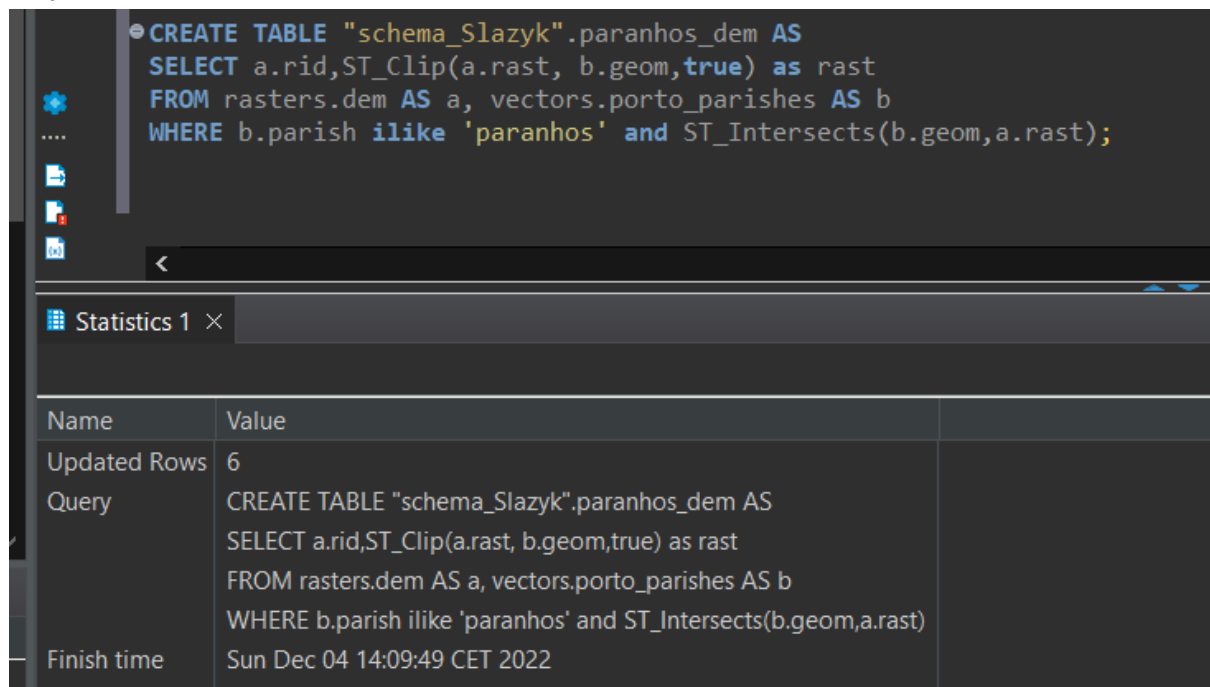
Przykład 1 - ST_Band



The screenshot shows a SQL query in a dark-themed IDE. The query creates a table named 'landsat_nir' in the 'schema_Slazyk' schema, selecting 'rid' and 'ST_Band(rast,4)' as 'rast' from the 'rasters.dem' table. Below the query editor, a 'Statistics 1' window is open, displaying the following data:

Name	Value
Updated Rows	630
Query	CREATE TABLE "schema_Slazyk".landsat_nir AS SELECT rid, ST_Band(rast,4) AS rast FROM rasters.dem
Finish time	Sun Dec 04 14:08:19 CET 2022

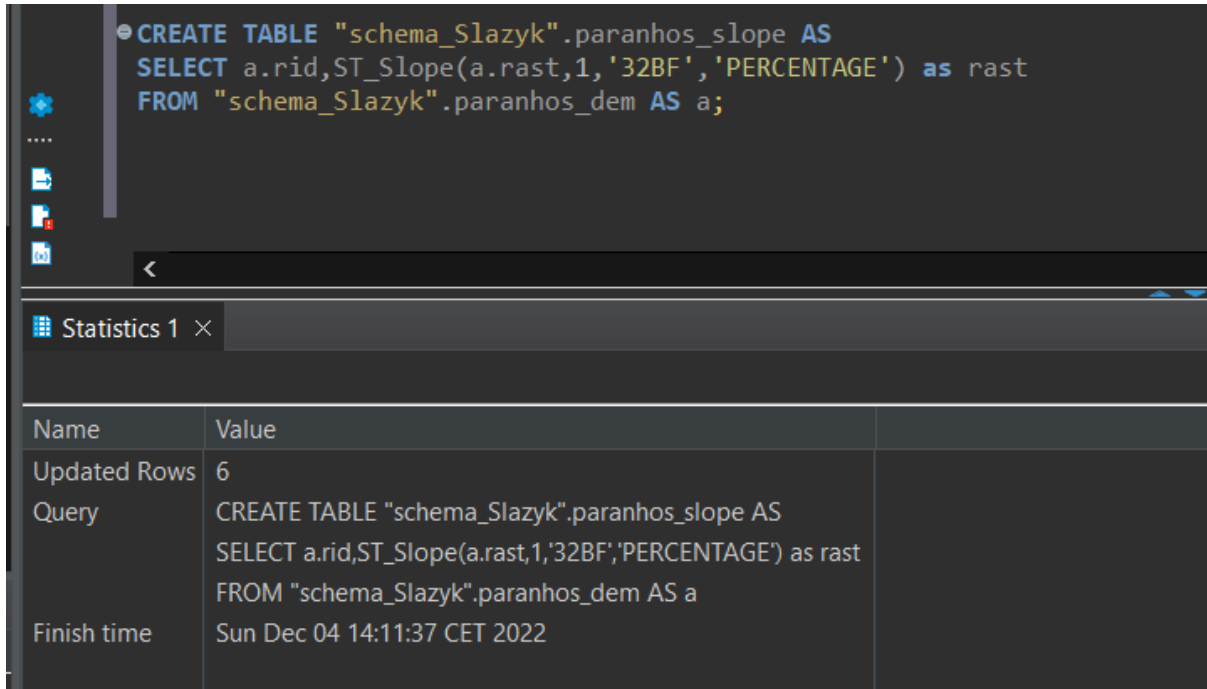
Przykład 2 - ST_Clip



The screenshot shows a SQL query in a dark-themed IDE. The query creates a table named 'paranhos_dem' in the 'schema_Slazyk' schema, selecting 'a.rid', 'ST_Clip(a.rast, b.geom, true)' as 'rast' from the 'rasters.dem' table (aliased as 'a') and 'vectors.porto_parishes' table (aliased as 'b'). The query includes a WHERE clause: 'b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast)'. Below the query editor, a 'Statistics 1' window is open, displaying the following data:

Name	Value
Updated Rows	6
Query	CREATE TABLE "schema_Slazyk".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)
Finish time	Sun Dec 04 14:09:49 CET 2022

Przykład 3 - ST_Slope

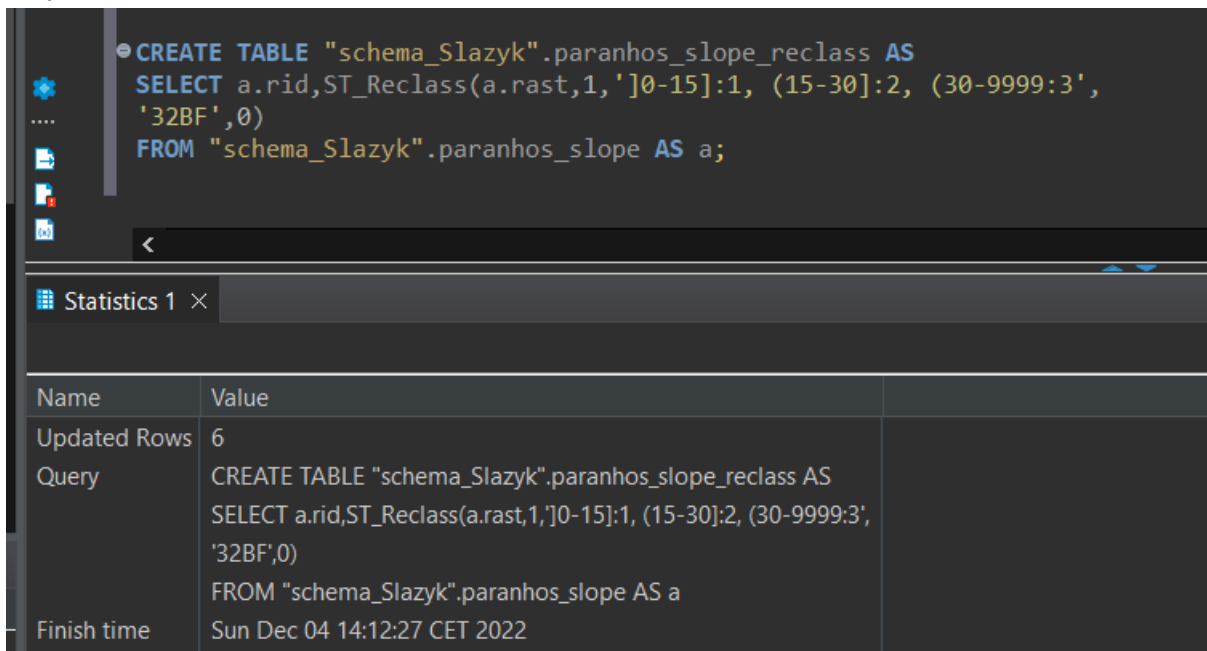


The screenshot shows a GIS application window with a dark theme. At the top, a SQL query is entered in a text area. Below the query, a 'Statistics 1' panel is open, displaying execution details. The query uses the ST_Slope function to calculate the slope of a raster dataset.

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

Name	Value
Updated Rows	6
Query	CREATE TABLE "schema_Slazyk".paranhos_slope AS SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast FROM "schema_Slazyk".paranhos_dem AS a
Finish time	Sun Dec 04 14:11:37 CET 2022

Przykład 4 - ST_Reclass



The screenshot shows the same GIS application window as in the previous example. The SQL query now uses the ST_Reclass function to reclassify a raster dataset based on three value ranges. The 'Statistics 1' panel shows the execution results.

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

Name	Value
Updated Rows	6
Query	CREATE TABLE "schema_Slazyk".paranhos_slope_reclass AS SELECT a.rid,ST_Reclass(a.rast,1,['0-15]:1, (15-30]:2, (30-9999:3', '32BF',0) FROM "schema_Slazyk".paranhos_slope AS a
Finish time	Sun Dec 04 14:12:27 CET 2022

Przykład 5 - ST_SummaryStats

SQL Query:

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

Results 1 ×

SQL Filter: `SELECT st_summarystats(a.rast) AS stats FROM "schema_Slazyk".paranhos_dem AS a;` Enter a SQL expression to filter results (use Ctrl+S)

	stats						
	count	sum	mean	stddev	min	max	
1	201	2,125,431	10,574.2835820896	1,300.4928741062	8,259	16,320	
2	727	7,509,266	10,329.1141678129	1,343.3366982639	7,228	14,861	
3	337	3,427,187	10,169.6943620178	1,584.0295979304	6,834	16,529	
4	34	355,473	10,455.0882352941	701.3987440086	9,278	12,098	
5	4,267	45,261,151	10,607.2535739395	1,283.6466423885	7,074	18,620	
6	2,395	26,501,357	11,065.2847599165	1,321.4788511304	7,997	19,919	

Przykład 6 - ST_SummaryStats oraz Union

SQL Query:

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

Results 1 ×

SQL Filter: `SELECT st_summarystats(ST_Union(a.rast)) FROM "schema_Slazyk".paranhos_dem AS a;` Enter a SQL expression to filter results (use Ctrl+S)

	st_summarystats						
	count	sum	mean	stddev	min	max	
1	7,961	85,179,865	10,699.6438889587	1,339.4848543728	6,834	19,919	

Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych

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

Results 1 ×

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

	123 min	123 max	123 mean
1	6,834	19,919	10,699.6438889587

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;
```

porto_parishes 1 ×

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;

	abc parish	123 min	123 max	123 mean
1	Bonfim	6,369	21,361	10,416.2133100843
2	Campanhã	6,319	24,149	9,777.0685989257
3	Paranhos	6,834	19,919	10,699.6438889587
4	Ramalde	6,826	31,920	10,744.4076019777
5	União das freguesias de Aldoar, Foz do Douro e Nevogilde	7,347	22,829	13,971.5903077366
6	União das freguesias de Cedofeita, Santo Ildefonso, Sé, Miragaia, São Nicolau e Vitória	6,486	21,438	10,833.142004971
7	União das freguesias de Lordelo do Ouro e Massarelos	6,477	22,358	12,187.9001612903

Przykład 9 - ST_Value

The screenshot shows a GIS application window. At the top, a SQL query is entered in a text area:

```
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;
```

Below the query, a tab labeled "places 1" is active. A filter bar shows the same query: `SELECT b.name, st_value(a.rast, (ST_Dump(b.geom)).geom)`. Below the filter bar is a table with two columns: "name" and "st_value". The table contains 13 rows of data, sorted by name. The "st_value" column contains numerical values representing elevation.

	name	st_value
1	Aldeia São Miguel	8,922
2	Alpendurada e Matos	9,533
3	Amarante	8,238
4	Baião	7,273
5	Cabeceiras de Basto	8,411
6	Castelo de Paiva	8,761
7	Celorico de Basto	9,419
8	Cinfães	8,699
9	Espinho	16,708
10	Fafe	10,838
11	Fajozes	9,512
12	Felgueiras	10,846
13	Gondomar	9,764

Topographic Position Index (TPI)

Przykład 10 - ST_TPI

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

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

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

Results 1 ×

ST_TPI AddRasterConstraints('schema_Slazyk'::name, 'tpi30'::name, 'rast'::name) Enter a SQL expression to filter results (use Ctrl+Space)

Grid	addrasterconstraints	Value
1	[v]	true

Algebra map

Przykład 1 - Wyrażenie Algebra Map

```
CREATE TABLE "schema_Slazyk".porto_ndvi AS
WITH r 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.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_Slazyk".porto_ndvi
USING gist (ST_ConvexHull(rast));

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

Results 1 ×

ST_MapAlgebra AddRasterConstraints('schema_Slazyk'::name, 'porto_ndvi'::name, 'rast'::name) Enter a SQL expression to filter results (use Ctrl+Space)

Grid	addrasterconstraints	Value
1	[v]	true

Przykład 2 – Funkcja zwrotna

```
create or replace function "schema_Slazyk".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;
```

Statistics 1 ×

Name	Value
Updated Rows	0
Query	create or replace function "schema_Slazyk".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

Eksport danych

Przykład 1 - ST_AsTiff

```
SELECT ST_AsTiff(ST_Union(rast))  
FROM "schema_Slazyk".porto_ndvi;
```

Results 1 ×


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

Grid	st_astiff
1	* ... [275539]

Przykład 2 - ST_AsGDALRaster

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

Results 1 ×

SELECT ST_AsGDALRaster(ST_Union(rast), 'GTiff') |  Enter a SQL expression to filter results (use Ctrl+Space)

	st_asgdalraster	
1	* ³ ... [148838]	

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 "schema_Slazyk".porto_ndvi;

SELECT lo_export(loid, 'D:\Studia\SEM7\BDP\myraster.tiff') FROM tmp_out;
```

Results 1 ×

SELECT lo_export(loid, 'D:\Studia\SEM7\myraste

Grid	123 lo_export
1	1

