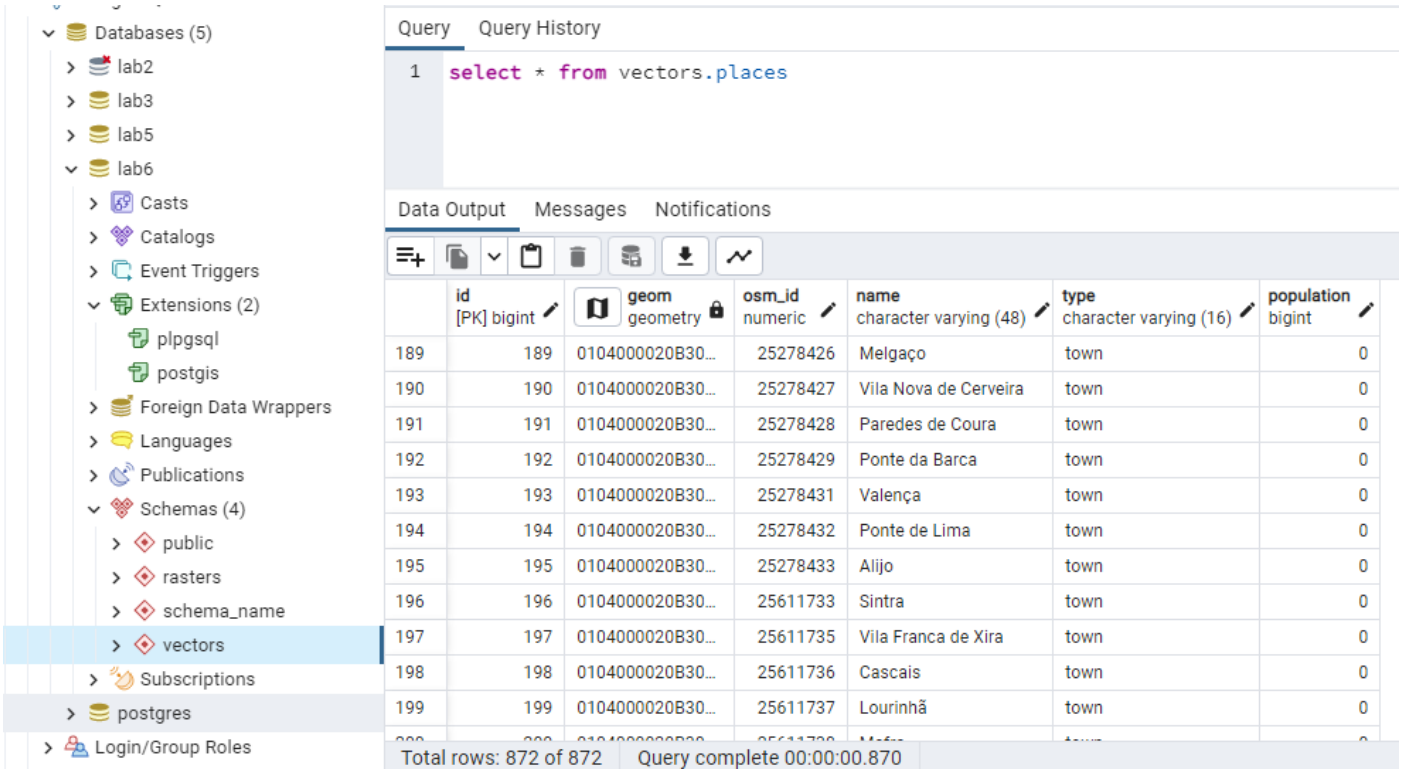
Załadowanie zawartości bazy danych postgis_raster.backup



Ładowanie zawartości plików

```
C:\Program Files\PostgreSQL\15\bin>raster2pgsql.exe -s 3763 -N -32767 -t 128x128 -I -C -M -d D:\Semestr_7\BazyDanychPrzestrzennych\lab6\rasters\Landsat8_L1TP_RGBN.TIF rasters.landsat8 | psql -d lab6 -h localhost -U postgres -p 5432
Processing 1/1: D:\Semestr_7\BazyDanychPrzestrzennych\lab6\rasters\Landsat8_L1TP_RGBN.TIF
Password for user postgres:
BEGIN
NOTICE:  table "landsat8" does not exist, skipping
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
INSERT 0 1
```

```
INSERT 0 1
INSERT 0 1
CREATE INDEX
ANALYZE
NOTICE:  Adding SRID constraint
NOTICE:  Adding scale-X constraint
NOTICE:  Adding scale-Y constraint
NOTICE:  Adding blocksize-X constraint
NOTICE:  Adding blocksize-Y constraint
NOTICE:  Adding alignment constraint
NOTICE:  Adding number of bands constraint
NOTICE:  Adding pixel type constraint
NOTICE:  Adding nodata value constraint
NOTICE:  Adding out-of-database constraint
NOTICE:  Adding maximum extent constraint
 addrasterconstraints
----------------------
 t
(1 row)

COMMIT
VACUUM
```

```
C:\Program Files\PostgreSQL\15\bin>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d D:\Semestr_7\BazyDanychPrze
strzennych\lab6\rasters\srtm_1arc_v3.tif rasters.dem | psql -d lab6 -h localhost -U postgres -p 5432
Processing 1/1: D:\Semestr_7\BazyDanychPrzestrzennych\lab6\rasters\srtm_1arc_v3.tif
Password for user postgres:
BEGIN
NOTICE:  table "dem" does not exist, skipping
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
```

```
INSERT 0 1
INSERT 0 1
INSERT 0 1
CREATE INDEX
ANALYZE
NOTICE:  Adding SRID constraint
NOTICE:  Adding scale-X constraint
NOTICE:  Adding scale-Y constraint
NOTICE:  Adding blocksize-X constraint
NOTICE:  Adding blocksize-Y constraint
NOTICE:  Adding alignment constraint
NOTICE:  Adding number of bands constraint
NOTICE:  Adding pixel type constraint
NOTICE:  Adding nodata value constraint
NOTICE:  Adding out-of-database constraint
NOTICE:  Adding maximum extent constraint
 addrasterconstraints
----------------------
 t
(1 row)

COMMIT
VACUUM

C:\Program Files\PostgreSQL\15\bin>
```

Tworzenie rastrów z istniejących rastrów.

Przykład 1

Query | Query History

```
1  CREATE TABLE stepniewski.intersects AS
2  SELECT a.rast, b.municipality
3  FROM rasters.dem AS a, vectors.porto_parishes AS b
4  WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto';
5
```

Data Output | Messages | Notifications

```
SELECT 25

Query returned successfully in 156 msec.
```

Query Query History

```
1  alter table stepniewski.intersects
2  add column rid SERIAL PRIMARY KEY;
3
```

Data Output Messages Notifications

```
ALTER TABLE

Query returned successfully in 84 msec.
```

Query Query History

```
1  CREATE INDEX idx_intersects_rast_gist ON stepniewski.intersects
2  USING gist (ST_ConvexHull(rast));
3
```

Data Output Messages Notifications

```
CREATE INDEX

Query returned successfully in 59 msec.
```

Query Query History

```
1  SELECT AddRasterConstraints('stepniewski'::name,
2  'intersects'::name,'rast'::name);
3
```

Data Output Messages Notifications

	addrasterconstraints boolean
1	true

Przykład 2

Query Query History

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

Data Output Messages Notifications

```
SELECT 25

Query returned successfully in 153 msec.
```

Przykład 3

```sql
CREATE TABLE stepniewski.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);
```

Data Output | Messages | Notifications

```
SELECT 1

Query returned successfully in 97 msec.
```

Tworzenie rastrów z wektorów (rastrowanie)

Przykład 1

Query | Query History

```sql
CREATE TABLE stepniewski.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';
```

Data Output | Messages | Notifications

```
SELECT 7

Query returned successfully in 83 msec.
```

Przykład 2

Query | Query History

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

Data Output | Messages | Notifications

```
SELECT 1

Query returned successfully in 91 msec.
```

Przykład 3

Query | Query History

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

Data Output | Messages | Notifications

```
SELECT 8

Query returned successfully in 112 msec.
```

Konwertowanie rastrów na wektory (wektoryzowanie)

Przykład 1

Query | Query History

```
1  create table stepniewski.intersection as
2  SELECT
3  a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast)
4  ).val
5  FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
6  WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Data Output | Messages | Notifications

```
SELECT 6629

Query returned successfully in 3 secs 667 msec.
```

Przykład 2

Query | Query History

```
1  CREATE TABLE stepniewski.dumppolygons AS
2  SELECT
3  a.rid,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).geom, (ST_DumpAsPolygons(ST_Clip(a.ra
4  FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
5  WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Data Output | Messages | Notifications

```
SELECT 6422

Query returned successfully in 295 msec.
```

Analiza rastrów

Przykład 1

```
1  CREATE TABLE stepniewski.landsat_nir AS
2  SELECT rid, ST_Band(rast,4) AS rast
3  FROM rasters.landsat8;
```

Data Output | Messages | Notifications

```
SELECT 384

Query returned successfully in 860 msec.
```

Przykład 2

Query | Query History

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

Data Output | Messages | Notifications

```
SELECT 4

Query returned successfully in 113 msec.
```

Przykład 3

Query | Query History

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

Data Output | Messages | Notifications

```
SELECT 4

Query returned successfully in 661 msec.
```

Przykład 4

Query | Query History

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

Data Output | Messages | Notifications

```
SELECT 4

Query returned successfully in 100 msec.
```

Przykład 5

Query Query History

```
1  SELECT st_summarystats(a.rast) AS stats
2  FROM stepniewski.paranhos_dem AS a;
```

Data Output Messages Notifications

	stats summarystats
1	(2616,278385,...
2	(682,95581,14...
3	(216,31874,14...
4	(6463,816615,...

Przykład 6

```
1  SELECT st_summarystats(ST_Union(a.rast))
2  FROM stepniewski.paranhos_dem AS a;
```

Data Output Messages Notifications

	st_summarystats summarystats
1	(9977,1222455,1...

Przykład 7

```
1  WITH t AS (
2  SELECT st_summarystats(ST_Union(a.rast)) AS stats
3  FROM stepniewski.paranhos_dem AS a
4  )
5  SELECT (stats).min,(stats).max,(stats).mean FROM t;
```

Data Output Messages Notifications

	min double precision	max double precision	mean double precision
1	87	158	122.527312819484

Przykład 8

Query | Query History

Execute/Refresh F5

```
1  WITH t AS (
2      SELECT b.parish AS parish, st_summarystats(ST_Union(ST_Clip(a.rast,b.geom,true))) A
3      FROM rasters.dem AS a, vectors.porto_parishes AS b
4      WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
5      group by b.parish
6  )
7  SELECT parish,(stats).min,(stats).max,(stats).mean FROM t;
```

Data Output | Messages | Notifications

	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 d...	-4	83	34.66735489791237
6	União das freguesias d...	1	157	95.00277741039545
7	União das freguesias d...	-1	117	49.50051440329218

Przykład 9

Query | Query History

```
1  SELECT b.name,st_value(a.rast,(ST_Dump(b.geom)).geom)
2  FROM rasters.dem a, vectors.places AS b
3  WHERE ST_Intersects(a.rast,b.geom)
4  ORDER BY b.name;
```

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 451 msec.
33 rows affected.
```

Przykład 10

Query | Query History

```
1  create table stepniewski.tpi30 as
2  select ST_TPI(a.rast,1) as rast
3  from rasters.dem a;
```

Data Output | Messages | Notifications

```
SELECT 589

Query returned successfully in 3 min 55 secs.
```

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

Data Output | Messages | Notifications

```
CREATE INDEX

Query returned successfully in 380 msec.
```

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

Data Output | Messages | Notifications

```
NOTICE:  Adding SRID constraint
NOTICE:  Adding scale-X constraint
NOTICE:  Adding scale-Y constraint
NOTICE:  Adding blocksize-X constraint
NOTICE:  Adding blocksize-Y constraint
NOTICE:  Adding alignment constraint
NOTICE:  Adding number of bands constraint
NOTICE:  Adding pixel type constraint
NOTICE:  Adding nodata value constraint
NOTICE:  Adding out-of-database constraint
NOTICE:  Adding maximum extent constraint

Successfully run. Total query runtime: 1 secs 346 msec.
1 rows affected.
```

Algebra map

Przykład 1

```
CREATE TABLE stepniewski.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;
```

Data Output | Messages | Notifications

```
SELECT 23

Query returned successfully in 852 msec.
```

Query Query History

```
1  CREATE INDEX idx_porto_ndvi_rast_gist ON stepniewski.porto_ndvi
2  USING gist (ST_ConvexHull(rast));
```

Data Output Messages Notifications

```
CREATE INDEX

Query returned successfully in 116 msec.
```

Query Query History

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

Data Output Messages Notifications

	addrasterconstraints boolean
1	true

Przykład 2

Query Query History

```
1  create or replace function stepniewski.ndvi(
2  value double precision [] [] [],
3  pos integer [][],
4  VARIADIC userargs text []
5  )
6  RETURNS double precision AS
7  $$
8  BEGIN
9  --RAISE NOTICE 'Pixel Value: %', value [1][1][1];-->For debug purposes
10 RETURN (value [2][1][1] - value [1][1][1])/(value [2][1][1]+value
11 [1][1][1]); --> NDVI calculation!
12 END;
13 $$
14 LANGUAGE 'plpgsql' IMMUTABLE COST 1000;
```

Data Output Messages Notifications

```
CREATE FUNCTION

Query returned successfully in 164 msec.
```

```sql
CREATE TABLE stepniewski.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],
'stepniewski.ndvi(double precision[],integer[],text[])'::regprocedure --> This is the f
) AS rast
FROM r;
```

Data Output | Messages | Notifications

```
SELECT 23

Query returned successfully in 606 msec.
```

```sql
CREATE INDEX idx_porto_ndvi2_rast_gist ON stepniewski.porto_ndvi2
USING gist (ST_ConvexHull(rast));
```

Data Output | Messages | Notifications

```
CREATE INDEX

Query returned successfully in 247 msec.
```

```sql
SELECT AddRasterConstraints('stepniewski'::name, 'porto_ndvi2'::name,'rast'::name);
```

Data Output | Messages | Notifications

	addrasterconstraints boolean
1	true

Eksport danych

Przykład 1

```sql
SELECT ST_AsTiff(ST_Union(rast))
FROM stepniewski.porto_ndvi;
```

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 316 msec.
1 rows affected.
```

Przykład 2

```
SELECT ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE','PREDICTOR=2',
FROM stepniewski.porto_ndvi;
```

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 312 msec.
1 rows affected.
```

Przykład 3

```
CREATE TABLE tmp_out AS
SELECT lo_from_bytea(0,
ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE','PREDICTOR=2', 'PZLEV
) AS loid
FROM stepniewski.porto_ndvi;
```

Data Output | Messages | Notifications

```
SELECT 1

Query returned successfully in 287 msec.
```

Query | Query History

```
SELECT lo_export(loid, 'D:\Semestr_7\BazyDanychPrzestrzennych\lab6\myraster.tiff') -->
FROM tmp_out;
```

Data Output | Messages | Notifications

	lo_export integer
1	1

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

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 330 msec.
1 rows affected.
```