

Nowa baza danych

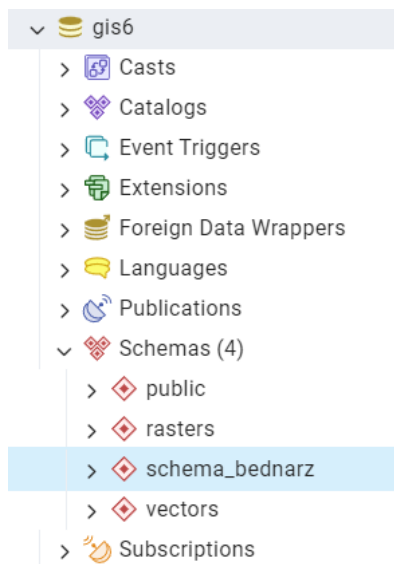
Pierwszym krokiem warsztatów jest stworzenie nowej bazy oraz załadowanie do niej kopii bazy danych zapisanej w pliku *postgis_raster.backup* i zawierającej niezbędną strukturę wraz z danymi.

Struktura bazy danych

Baza danych ma następującą strukturę:

- `schema_name` - (zmień nazwę tego schematu na swoje nazwisko). Schemat jest pusty, będzie on zawierać wyniki Twoich zadań
- `public`
- `rasters` - schemat jest pusty, będzie on zawierać wyniki Twoich zadań
- `vectors`
 - `railroad` - linie
 - `places` - punkty
 - `porto_parishes` - poligony

Tabela `vectors.porto_parishes` zawiera dane parafii z okolic Porto w Portugalii.



```
C:\Program Files\PostgreSQL\16\bin>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d "C:\Users\kamil\Desktop\PostGIS raster - dane\PostGIS raster - dane\srtm_larc_v3.tif" rasters.dem > "C:\Users\kamil\Desktop\PostGIS raster - dane\PostGIS raster - dane\dem.sql"
Processing 1/1: C:\Users\kamil\Desktop\PostGIS raster - dane\PostGIS raster - dane\srtm_larc_v3.tif
```

Przykład 3 – załadowanie danych landsat 8 o wielkości kafelka 128x128 bezpośrednio do bazy danych.

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

Po wykonaniu powyższych przykładów sprawdź schemat rasters w Twojej bazie danych. Sprawdź strukturę oraz zawartość widoku public.raster_columns.

	r_table_catalog name	r_table_schema name	r_table_name name	r_raster_column name	srid integer	scale_x double precision	scale_y double precision	blocksize_x integer	blocksize_y integer	same_alignment boolean	regular_t boolean
1	gis6	rasters	landsat8	rast	3763	30.3114020783	-29.7057939174	128	128	true	false
2	gis6	rasters	dem	rast	3763	23.3527411668	-30.7891756029	100	100	true	false

Tworzenie rastrow z istniejących rastrow i interakcja z wektorami

Przykład 1 - ST_Intersects

```
6  -- ST_Intersects
7  ✓ CREATE TABLE schema_bednarz.intersects AS
8  SELECT a.rast, b.municipality
9  FROM rasters.dem AS a, vectors.porto_parishes AS b
10 WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'por
11
12 -- dodanie serial primary key
13 ✓ alter table schema_bednarz.intersects
14 add column rid SERIAL PRIMARY KEY;
15
16 -- utworzenie indeksu przestrzennego
17 ✓ CREATE INDEX idx_intersects_rast_gist ON schema_bednarz.intersect
18 USING gist (ST_ConvexHull(rast));
19
20 -- dodanie raster constraints
21 -- schema::name table_name::name raster_column::name
22 ✓ SELECT AddRasterConstraints('schema_bednarz'::name,
23 'intersects'::name, 'rast'::name);
24
25 SELECT * FROM schema_bednarz.intersects;
```

Data Output Messages Notifications

		SQL	
		rast raster	
1		01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0474F11FE054A04410000	
2		01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C082283B553DAA0441000C	

Przykład 2 – ST_Clip

```
27 -- ST_Clip
28 ✓ CREATE TABLE schema_bednarz.clip AS
29 SELECT ST_Clip(a.rast, b.geom, true), b.municipality
30 FROM rasters.dem AS a, vectors.porto_parishes AS b
31 WHERE ST_Intersects(a.rast, b.geom) AND b.municipality like 'PORTO';
```

Data Output Messages Notifications

SELECT 25

Query returned successfully in 87 msec.

Przykład 3 – ST_Union

```
-- ST_Union
✓ CREATE TABLE schema_bednarz.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 * FROM schema_bednarz.union;
```

Data Output Messages Notifications

st_union
raster
01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C0DAE35DC0089604410000000000000000

Tworzenie rastrow z wektorów (rastrowanie)

Przykład 1 - ST_AsRaster

```
43 -- Tworzenie rastrow z wektorów (rastrowanie)
44 -- Przykład 1 - ST_AsRaster
45 ✓ CREATE TABLE schema_bednarz.porto_parishes AS
46 WITH r AS (
47     SELECT rast FROM rasters.dem
48     LIMIT 1
49 )
50 SELECT ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767) AS rast
51 FROM vectors.porto_parishes AS a, r
52 WHERE a.municipality ilike 'porto';
53
54 SELECT * FROM schema_bednarz.porto_parishes;
```

Data Output Messages Notifications

	rast
	raster
1	01000001006172BF3E4D5A374080318D6907CA3EC0574768B43454E3C0D0979D709B540
2	01000001006172BF3E4D5A374080318D6907CA3EC00A62D67B5CE2E2C08A6D93EAF7DC
3	01000001006172BF3E4D5A374080318D6907CA3EC0D268172037B9E5C0CA65352F39910
4	01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C05E060A8BE77F0

Przykład 2 - ST_Union

```
56 -- Przykład 2 - ST_Union
57 DROP TABLE schema_bednarz.porto_parishes; --> drop table porto_parishes first
58 CREATE TABLE schema_bednarz.porto_parishes AS
59 WITH r AS (
60     SELECT rast FROM rasters.dem
61     LIMIT 1
62 )
63 SELECT st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767)) AS rast
64 FROM vectors.porto_parishes AS a, r
65 WHERE a.municipality ilike 'porto';
66
67 SELECT * FROM schema_bednarz.porto_parishes;
```

Data Output Messages Notifications

SQL

	rast raster
1	01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C0DAE35DC008960441000000000000000000000

Przykład 3 - ST_Tile

```
69 -- Przykład 3 - ST_Tile
70 DROP TABLE schema_bednarz.porto_parishes; --> drop table porto_parishes first
71 CREATE TABLE schema_bednarz.porto_parishes AS
72 WITH r AS (
73     SELECT rast FROM rasters.dem
74     LIMIT 1 )
75 SELECT st_tile(st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,
76 32767)),128,128,true,-32767) AS rast
77 FROM vectors.porto_parishes AS a, r
78 WHERE a.municipality ilike 'porto';
79
80 SELECT * FROM schema_bednarz.porto_parishes;
```

Data Output Messages Notifications

SQL

	rast raster
1	01000001006172BF3E4D5A374080318D6907CA3EC0D6994F2F0BE0E6C0DAE35DC008960441000000000000000000000
2	01000001006172BF3E4D5A374080318D6907CA3EC0B0A2635B666AE5C0DAE35DC008960441000000000000000000000
3	01000001006172BF3E4D5A374080318D6907CA3EC08AAB7787C1F4E3C0DAE35DC008960441000000000000000000000
4	01000001006172BF3E4D5A374080318D6907CA3EC064B48BB31C7FE2C0DAE35DC0089604410000000000000000000000

Konwertowanie rastrow na wektory (wektoryzowanie)

Przykład 1 - ST_Intersection

```
83 -- Konwertowanie rastrow na wektory (wektoryzowanie)
84 -- Przykład 1 - ST_Intersection
85 CREATE TABLE schema_bednarz.intersection AS
86 SELECT
87     a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast)
88     ).val
89 FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
90 WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Data Output Messages Notifications

SELECT 6629

Query returned successfully in 5 secs 106 msec.

Przykład 2 - ST_DumpAsPolygons

```
91
92 -- Przykład 2 - ST_DumpAsPolygons
93 CREATE TABLE schema_bednarz.dumppolygons AS
94 SELECT
95 a.rid,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).geom,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).v
96 FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
97 WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
```

Data Output Messages Notifications

SELECT 6422

Query returned successfully in 221 msec.

Analiza rastrów

Przykład 1 - ST_Band

```
100 -- Analiza rastrów
101 -- Przykład 1 - ST_Band
102 CREATE TABLE schema_bednarz.landsat_nir AS
103 SELECT rid, ST_Band(rast,4) AS rast
104 FROM rasters.landsat8;
105
106 SELECT * FROM schema_bednarz.landsat_nir;
```

Data Output Messages Notifications

	rid integer		rast raster
1	1		01000001003849EE0BB84F3E404F2001E9AEB43DC02068193DBB
2	2		01000001003849EE0BB84F3E404F2001E9AEB43DC08C835ABCBF
3	3		01000001003849EE0BB84F3E404F2001E9AEB43DC0F99E9B3BC4

Total rows: 384 of 384 Query complete 00:00:01.166 Ln 106, Col 1

Przykład 2 - ST_Clip

```
108 -- Przykład 2 - ST_Clip
109 CREATE TABLE schema_bednarz.paranhos_dem AS
110 SELECT a.rid,ST_Clip(a.rast, b.geom,true) as rast
111 FROM rasters.dem AS a, vectors.porto_parishes AS b
112 WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
113
114 SELECT * FROM schema_bednarz.paranhos_dem;
```

Data Output Messages Notifications

	rid integer		rast raster
1	380		01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2F9204
2	382		01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6CAD850
3	412		01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE054A04

Przykład 3 - ST_Slope

```
116 -- Przykład 3 - ST_Slope
117 CREATE TABLE schema_bednarz.paranhos_slope AS
118 SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
119 FROM schema_bednarz.paranhos_dem AS a;
120
121 SELECT * FROM schema_bednarz.paranhos_slope;
```

Data Output			Messages	Notifications
	rid integer	rast raster		
1	380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707		
2	382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D		
3	412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11F		

Przykład 4 - ST_Reclass

```
123 -- Przykład 4 - ST_Reclass
124 CREATE TABLE schema_bednarz.paranhos_slope_reclass AS
125 SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3',
126 '32BF',0)
127 FROM schema_bednarz.paranhos_slope AS a;
128
129 SELECT * FROM schema_bednarz.paranhos_slope_reclass;
```

Data Output			Messages	Notifications
	rid integer	st_reclass raster		
1	380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F20441		
2	382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6CAD85044		

Przykład 5 - ST_SummaryStats

```
130 -- Przykład 5 - ST_SummaryStats
131 SELECT st_summarystats(a.rast) AS stats
132 FROM schema_bednarz.paranhos_dem AS a;
```

Data Output		Messages	Notifications
	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)		

Przykład 6 - ST_SummaryStats oraz Union

```
134 -- Przykład 6 - ST_SummaryStats oraz Union
135 SELECT st_summarystats(ST_Union(a.rast))
136 FROM schema_bednarz.paranhos_dem AS a;
137
```

Data Output		Messages	Notifications
	st_summarystats summarystats		
1	(9977,1222455,122.52731281948482,16.908004202736272,87,15...		

Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych

```
138 -- Przykład 7 - ST_SummaryStats z lepszą kontrolą złożonego typu danych
139 WITH t AS (
140     SELECT st_summarystats(ST_Union(a.rast)) AS stats
141     FROM schema_bednarz.paranhos_dem AS a
142 )
143 SELECT (stats).min,(stats).max,(stats).mean FROM t;
```

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

Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY

```
145 -- Przykład 8 - ST_SummaryStats w połączeniu z GROUP BY
146 WITH t AS (
147     SELECT b.parish AS parish, st_summarystats(ST_Union(ST_Clip(a.rast, b.geom,true))) AS stats
148     FROM rasters.dem AS a, vectors.porto_parishes AS b
149     WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
150     GROUP BY b.parish
151 )
152 SELECT parish,(stats).min,(stats).max,(stats).mean FROM t;
```

	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

Przykład 9 - ST_Value

```
154 -- Przykład 9 - ST_Value
155 SELECT b.name,st_value(a.rast,(ST_Dump(b.geom)).geom)
156 FROM rasters.dem a, vectors.places AS b
157 WHERE ST_Intersects(a.rast,b.geom)
158 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

Przykład 10 - ST_TPI

```
160 -- Przykład 10 - ST_TPI
161 create table schema_bednarz.tpi30 as
162 select ST_TPI(a.rast,1) as rast
163 from rasters.dem a;
164
165 CREATE INDEX idx_tpi30_rast_gist ON schema_bednarz.tpi30
166 USING gist (ST_ConvexHull(rast));
167
168 SELECT AddRasterConstraints('schema_bednarz'::name,
169 'tpi30'::name,'rast'::name);
```

	addrasterconstraints boolean
1	true

Problem do samodzielnego rozwiązania

```
171 -- Problem do samodzielnego rozwiązania
172 ✓ create table schema_bednarz.tpi30_porto as
173 SELECT ST_TPI(a.rast,1) as rast
174 FROM rasters.dem AS a, vectors.porto_parishes AS b
175 WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto';
176
177 ✓ CREATE INDEX idx_tpi30_porto_rast_gist ON schema_bednarz.tpi30_porto
178 USING gist (ST_ConvexHull(rast));
179
180 ✓ SELECT AddRasterConstraints('schema_bednarz'::name,
181 'tpi30_porto'::name, 'rast'::name);
```

Data Output Messages Notifications

addrasterconstraints						
boolean						
1	true					

Algebra map

Przykład 1 - Wyrażenie Algebra Map

```
✓ CREATE TABLE schema_bednarz.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'
  ) AS rast
FROM r;

✓ CREATE INDEX idx_porto_ndvi_rast_gist ON schema_bednarz.porto_ndvi
USING gist (ST_ConvexHull(rast));

✓ SELECT AddRasterConstraints('schema_bednarz'::name,
'porto_ndvi'::name, 'rast'::name);

SELECT * FROM schema_bednarz.porto_ndvi;
```

Output Messages Notifications

rid		rast				
integer		raster				
245	01000001003849EE0BB84F3E404F2001E9AEB43DC0086F60B09B56E3C01F1195122C32044100000					
270	01000001003849EE0BB84F3E404F2001E9AEB43DC075735DC2074FE3C0E4374673AE11044100000					
246	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C075C249A27255044100000					

Przykład 2 – Funkcja zwrotna

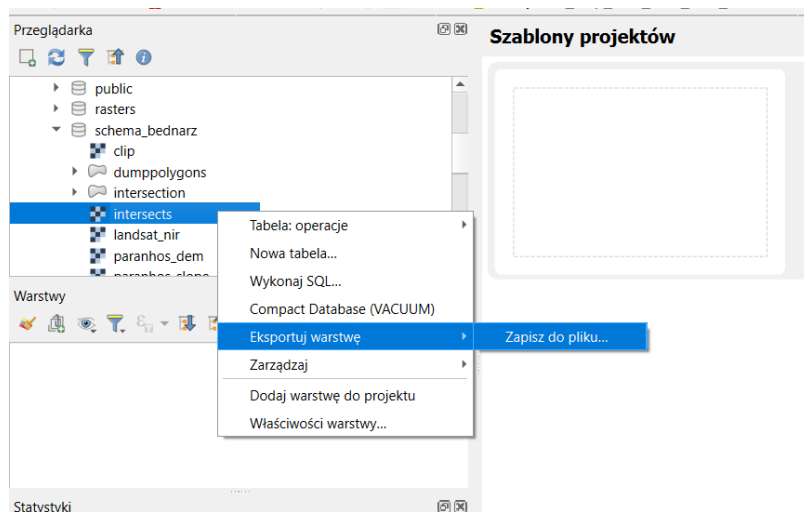
```
223 CREATE TABLE schema_bednarz.porto_ndvi2 AS
224 WITH r AS (
225     SELECT a.rid,ST_Clip(a.rast, b.geom,true) AS rast
226     FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
227     WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
228 )
229 SELECT
230     r.rid,ST_MapAlgebra(
231         r.rast, ARRAY[1,4],
232         'schema_bednarz.ndvi(double precision[], integer[],text[])::regproc',
233         '32BF'::text
234     ) AS rast
235 FROM r;
236
237 CREATE INDEX idx_porto_ndvi2_rast_gist ON schema_bednarz.porto_ndvi2
238 USING gist (ST_ConvexHull(rast));
239
240 SELECT AddRasterConstraints('schema_bednarz'::name,
241     'porto_ndvi2'::name,'rast'::name);
242
243 SELECT * FROM schema_bednarz.porto_ndvi2;
```

Data Output Messages Notifications

	rid integer	rast raster
1	245	01000001003849EE0BB84F3E404F2001E9AEB43DC0086F60B09B56E3C01F1195122C32044100000000

Eksport danych

Przykład 0 - Użycie QGIS



Przykład 1 - ST_AsTiff

```
246 -- Eksport danych
247 -- Przykład 1 - ST_AsTiff
248 v SELECT ST_AsTiff(ST_Union(rast))
249 FROM schema_bednarz.porto_ndvi;
```

Data Output Messages Notifications

	st_astiff	bytea
1	[binary dat...]	

Przykład 2 - ST_AsGDALRaster

```
250
251 -- Przykład 2 - ST_AsGDALRaster
252 v SELECT ST_AsGDALRaster(ST_Union(rast), 'GTiff')
253 FROM schema_bednarz.porto_ndvi;
```

Data Output Messages Notifications

	st_asgdalraster	bytea
1	[binary data]	

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

```
255 -- Przykład 3 - Zapisywanie danych na dysku za pomocą dużego obiektu (large object, lo)
256 v CREATE TABLE tmp_out AS
257 SELECT lo_from_bytea(0,
258 ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE', 'PREDICTOR=2', 'PZLEVEL=9'])
259 ) AS loid
260 FROM schema_bednarz.porto_ndvi;
261 -----
262 v SELECT lo_export(loid, 'C:\Users\kamil\Desktop\PostGIS raster - dane\PostGIS raster - dane\myraster
263 -- where the user postgres have access. In windows a flash drive usually works fine.
264 FROM tmp_out; -----
265 v SELECT lo_unlink(loid)
266 FROM tmp_out; --> Delete the large object.
267
```

Data Output Messages Notifications

ERROR: could not create server file "C:\Users\kamil\Desktop\PostGIS raster - dane\PostGIS raster - dane\myraster.tiff": Permission denied

SQL state: 42501

Przykład 4 - Użycie Gdal

```
PS C:\Users\kamil> gdal_translate -co COMPRESS=DEFLATE -co PREDICTOR=2 -co ZLEVEL=9 PG:"host=localhost port=5432 dbname=gis6 user=postgres password=
schema=schema_name table=porto_ndvi mode=2" porto_ndvi.tiff
```

Publikowanie danych za pomocą MapServer

Przy budowaniu obrazu dockerowego wyrzuca taki błąd, więc nie męczyłem się z resztą zadania.

```
-----
- LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line
Dockerfile:54
-----
52 |     ADD setup.sh /setup.sh
53 |     RUN chmod 0755 /setup.sh
54 | >>> RUN /setup.sh
55 |
56 |
-----
ERROR: failed to solve: process "/bin/sh -c /setup.sh" did not complete successfully: exit code: 127
PS C:\Users\kamil\Desktop\PostGIS raster - dane\docker-mapserver> |
```