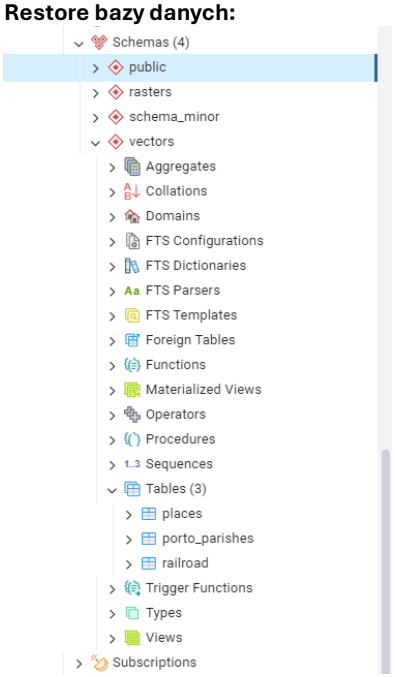
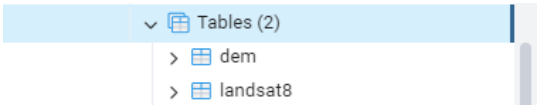
Karolina Minor
Bazy Danych Przestrzennych
Ćwiczenia 6-7

Restore bazy danych:



Wczytanie Landsat8_L1TP_RGBN.tif oraz srtm_1arc_v3.tif do rasters:



Data Output | Messages | Notifications

	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_blocking boolean	num_bands integer	pixel_types text[]
1	BDP6	rasters	dem	rast	3763	23.3527411668	-30.7891756029	100	100	true	false	1	{16BSI}
2	BDP6	rasters	landsat8	rast	3763	30.3114020783	-29.7057939174	128	128	true	false	4	{16BUI,16BUI,16BUI,16BUI}

Przykład 1 - ST_Intersects

Query | Query History

```
1  CREATE TABLE schema_minor.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
6  SELECT
7  ST_Width(rast),
8  ST_Height(rast),
9  ST_SRID(rast),
10 ST_NumBands(rast)
11 FROM schema_minor.intersects
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	100	100	3763	1
2	100	100	3763	1
3	100	100	3763	1
4	100	100	3763	1

Dodanie Primary Key:

Query | Query History

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

Dodanie indeksu przestrzennego:

Query | Query History

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

Data Output | Messages | Notifications

```
CREATE INDEX

Query returned successfully in 109 msec.
```

Przykład 2 - ST_Clip

Query | Query History

```
1 CREATE TABLE schema_minor.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 100 msec.
```

Query | Query History

```
1 SELECT
2 ST_Width(st_clip),
3 ST_Height(st_clip),
4 ST_SRID(st_clip),
5 ST_NumBands(st_clip)
6 FROM schema_minor.clip
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	66	93	3763	1
2	48	11	3763	1
3	17	30	3763	1
4	70	54	3763	1
5	100	87	3763	1
6	31	83	3763	1
7	15	32	3763	1

Przykład 3 - ST_Union

Query | Query History

```
1 CREATE TABLE schema_minor.union AS
2 SELECT ST_Union(ST_Clip(a.rast, b.geom, true))
3 FROM rasters.dem AS a, vectors.porto_parishes AS b
4 WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast);
```

Data Output | Messages | Notifications

```
SELECT 1

Query returned successfully in 91 msec.
```

Query | Query History

```
1 SELECT
2 ST_Width(st_union),
3 ST_Height(st_union),
4 ST_SRID(st_union),
5 ST_NumBands(st_union)
6 FROM schema_minor.union
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	498	172	3763	1

Tworzenie rastrów z wektorów - rastrowanie

Przykład 1 - ST_AsRaster

Query | Query History

```
1 CREATE TABLE schema_minor.porto_parishes AS
2 WITH r AS (
3 SELECT rast FROM rasters.dem
4 LIMIT 1
5 )
6 SELECT ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767) AS rast
7 FROM vectors.porto_parishes AS a, r
8 WHERE a.municipality ilike 'porto';
```

Data Output | Messages | Notifications

```
SELECT 7

Query returned successfully in 81 msec.
```

Query | Query History

```
1 SELECT
2 ST_Width(rast),
3 ST_Height(rast),
4 ST_SRID(rast),
5 ST_NumBands(rast)
6 FROM schema_minor.porto_parishes
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	66	105	3763	1
2	149	142	3763	1
3	146	90	3763	1
4	125	133	3763	1
5	202	88	3763	1
6	125	104	3763	1
7	171	87	3763	1

Przykład 2 - ST_Union

Query | Query History

```
DROP TABLE schema_minor.porto_parishes;
CREATE TABLE schema_minor.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 103 msec.
```

Query | Query History

```
SELECT
ST_Width(rast),
ST_Height(rast),
ST_SRID(rast),
ST_NumBands(rast)
FROM schema_minor.porto_parishes
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	499	173	3763	1

Przykład 3 - ST_Tile

Query | Query History

```
DROP TABLE schema_minor.porto_parishes;
CREATE TABLE schema_minor.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';
```

Data Output | Messages | Notifications

```
SELECT 8

Query returned successfully in 650 msec.
```

Query | Query History

```
SELECT
ST_Width(rast),
ST_Height(rast),
ST_SRID(rast),
ST_NumBands(rast)
FROM schema_minor.porto_parishes
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	128	128	3763	1
2	128	128	3763	1
3	128	128	3763	1
4	128	128	3763	1
5	128	128	3763	1
6	128	128	3763	1
7	128	128	3763	1
8	128	128	3763	1

Konwertowanie rastrów na wektory (wektoryzowanie)

Przykład 1 - ST_Intersection

Query | Query History

```
create table schema_minor.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);
```

Data Output | Messages | Notifications

```
SELECT 6629

Query returned successfully in 3 secs 223 msec.
```

Query | Query History

```
SELECT
rid,
ST_AsText(geom),
val
FROM schema_minor.intersection
```

Data Output | Messages | Notifications

	rid integer	st_astext text	val double precision
1	221	POLYGON((-39604.86528035818 168624.02902255123,-39633.587...	10648
2	221	POLYGON((-39574.553878279854 168624.02902255123,-39604.86...	12155
3	221	POLYGON((-39786.73369282809 168594.3232286338,-39794.4378...	9248
4	221	POLYGON((-39756.42229074977 168594.3232286338,-39786.7336...	10030
5	221	POLYGON((-39726.11088867145 168594.3232286338,-39756.4222...	10347
6	221	POLYGON((-39695.79948659313 168594.3232286338,-39726.1108...	10126
7	221	POLYGON((-39665.488084514815 168594.3232286338,-39695.799...	10611
8	221	POLYGON((-39635.1766824365 168564.61743471635,-39665.4880...	12761
9	221	POLYGON((-39604.86528035818 168624.02902255123,-39604.865...	14382

Przykład 2 - ST_DumpAsPolygons

Query | Query History

```
CREATE TABLE schema_minor.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);

```

Data Output | Messages | Notifications

```
SELECT 6422

Query returned successfully in 136 msec.
```

Query | Query History

```
SELECT
rid,
ST_AsText(geom),
val
FROM schema_minor.dumppolygons
```

Data Output | Messages | Notifications

	rid integer	st_astext text	val double precision
1	221	POLYGON((-39665.488084514815 168624.02902255123,-39665.488084514...	12761
2	221	POLYGON((-39635.17668243649 168624.02902255123,-39635.1766824364...	14382
3	221	POLYGON((-39604.86528035817 168624.02902255123,-39604.8652803581...	14090
4	221	POLYGON((-40089.84771361128 168594.3232286338,-40089.84771361128 ...	10625
5	221	POLYGON((-39907.979301141364 168594.3232286338,-39907.9793011413...	8601
6	221	POLYGON((-39817.0450949064 168594.3232286338,-39817.0450949064 16...	10102

Analiza Rastrów

Przykład 1 - ST_Band

Query | Query History

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

Query | Query History

```
SELECT
ST_Width(rast),
ST_Height(rast),
ST_SRID(rast),
ST_NumBands(rast)
FROM schema_minor.landsat_nir
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_srid integer	st_numbands integer
1	128	128	3763	1
2	128	128	3763	1
3	128	128	3763	1
4	128	128	3763	1
5	128	128	3763	1
6	128	128	3763	1

Przykład 2 - ST_Clip

Query | Query History

```
CREATE TABLE schema_minor.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);
```

Data Output | Messages | Notifications

```
SELECT 4

Query returned successfully in 69 msec.
```

Query | Query History

```
SELECT
ST_Width(rast),
ST_Height(rast),
ST_NumBands(rast)
FROM schema_minor.paranhos_dem
```

Data Output | Messages | Notifications

	st_width integer	st_height integer	st_numbands integer
1	53	73	1
2	49	32	1
3	59	8	1
4	100	79	1

Przykład 3 - ST_Slope]

Wygenerowanie nachylenia przy użyciu tabeli

Query | Query History

```
1 CREATE TABLE schema_minor.paranhos_slope AS
2 SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
3 FROM schema_minor.paranhos_dem AS a;
4
5 SELECT * FROM schema_minor.paranhos_slope
```

Data Output | Messages | Notifications

	rid integer	rast raster
1	380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2F92044100000000000000000000000
2	382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6CAD85044100000000000000000000000
3	412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE054A04410000000000000000000000
4	381	01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0DAE35DC00896044100000000000000000000000

Przykład 4 - ST_Reclass

Query | Query History

```
1 CREATE TABLE schema_minor.paranhos_slope_reclass AS
2 SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3',
3 '32BF',0)
4 FROM schema_minor.paranhos_slope AS a;
5
6 SELECT * FROM schema_minor.paranhos_slope
```

Data Output | Messages | Notifications

	rid integer	rast raster
1	380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2F92044100000000000000000000000
2	382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6CAD85044100000000000000000000000
3	412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE054A04410000000000000000000000
4	381	01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0DAE35DC00896044100000000000000000000000

Przykład 5 - ST_SummaryStats

Obliczanie statystyka rastra

Query | Query History

```
1 SELECT st_summarystats(a.rast) AS stats
2 FROM schema_minor.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)
4	(6463,816615,126.35231316725978,14.0438229209133,94,158)

Przykład 6 - ST_SummaryStats oraz Union

Wygenerowanie jednej statystyki wybranego rastra przy użyciu ST_Union

Query | Query History

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

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

Query | Query History

```
WITH t AS (
SELECT st_summarystats(ST_Union(a.rast)) AS stats
FROM schema_minor.paranhos_dem AS a
)
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	122.52731281948482

Przykład 8 - ST_SummaryStats w połączeniu z group by

Wyświetlanie statystyki dla każdego poligonu "parish"

Query | Query History

```
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;
```

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 de Aldoar, Foz do Douro e Nevogilde	-4	83	34.66735489791237
6	União das freguesias de Cedofeita, Santo Ildefonso, Sé, Miragaia, São Nicolau e Vitó…	1	157	95.00277741039545
7	União das freguesias de Lordelo do Ouro e Massarelos	-1	117	49.50051440329218

Przykład 9 - ST_Value

Przekonwertowanie geometrii wielopunktowej na geometrię jednopunktową za pomocą funkcji (ST_Dump(b.geom)).geom

```
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;
```

	name character varying (48)	st_value double precision
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	284

Przykład 10 - ST_TPI

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

SELECT st_summarystats(t.rast) FROM schema_minor.tpi30 AS t
```

	st_summarystats summarystats
1	(10000,0,0,0,0,0)
2	(10000,0,0,0.4727115928343625,-5.625,5)
3	(10000,0,0,0.7806607778030107,-5.25,5.125)
4	(10000,0,0,0.9911814289018942,-7.25,7)
5	(10000,0,0,1.512368795962151,-8.25,9.625)
6	(10000,0,0,1.4073467945037583,-9.75,12.625)

Utworzenie indeksu przestrzennego:

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

```
CREATE INDEX

Query returned successfully in 85 msec.
```

Dodanie constraintów:

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

	addrasterconstraints boolean
1	true

Problem do samodzielnego rozwiązania

Query | Query History

```
1  CREATE TABLE schema_minor.tpi30_porto AS
2  SELECT ST_TPI(a.rast, 1) AS rast
3  FROM rasters.dem AS a, vectors.porto_parishes AS b
4  WHERE ST_Intersects(a.rast, b.geom)
5    AND b.municipality ILIKE 'porto';
6
7  CREATE INDEX idx_tpi30_porto_rast_gist ON schema_minor.tpi30_porto
8  USING gist (ST_ConvexHull(rast));
9
10 SELECT AddRasterConstraints('schema_minor'::name,
11 'tpi30_porto'::name,'rast'::name);
```

Data Output | Messages | Notifications

	addrasterconstraints boolean
1	true

Porównanie czasów:

Query | Query History

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

Data Output | Messages | Notifications

	QUERY PLAN text
1	Seq Scan on dem a (cost=0.00..162.14 rows=589 width=32) (actual time=45.652..27669.281 rows=589 loops...
2	Planning Time: 0.178 ms
3	Execution Time: 29010.080 ms

Query | Query History

```
1  EXPLAIN ANALYZE
2  CREATE TABLE schema_minor.tpi30_porto AS
3  SELECT ST_TPI(a.rast, 1) AS rast
4  FROM rasters.dem AS a, vectors.porto_parishes AS b
5  WHERE ST_Intersects(a.rast, b.geom)
6    AND b.municipality ILIKE 'porto';
```

Data Output | Messages | Notifications

	QUERY PLAN text
1	Nested Loop (cost=0.14..205.98 rows=1 width=32) (actual time=50.153..1208.261 rows=25 loops=1)
2	-> Seq Scan on porto_parishes b (cost=0.00..147.04 rows=7 width=8358) (actual time=0.228..0.414 rows=7 loops=1)
3	Filter: ((municipality)::text ~~* 'porto'::text)
4	Rows Removed by Filter: 236
5	-> Index Scan using dem_st_convexhull_idx on dem a (cost=0.14..8.37 rows=1 width=88) (actual time=0.129..0.453 rows=4 loop...
6	Index Cond: ((rast)::geometry && b.geom)
7	Filter: _st_intersects(b.geom, rast, NULL::integer)
8	Rows Removed by Filter: 0
9	Planning Time: 0.211 ms
10	Execution Time: 1282.204 ms

Algebra map

Przykład 1 - Wyrażenie Algebry Map

Query | Query History

```
CREATE TABLE schema_minor.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;

SELECT st_summarystats(p.rast) FROM schema_minor.porto_ndvi AS p
```

Data Output | Messages | Notifications

	st_summarystats summarystats
1	(0,,,,,)
2	(1245,191.9515317317564,0.15417793713394087,0.11066626399581464,-0.05336048826575279,0.5319941639900208)
3	(2196,280.6113417702727,0.1277829425183391,0.06599777954104259,0.0012179126497358084,0.5395634770393372)
4	(1150,217.55006091190444,0.1891739660103517,0.1454306238620828,-0.05492142215371132,0.5156594514846802)
5	(270,80.92967846244574,0.2997395498609101,0.0981692238693096,0.08118022233247757,0.510151743888855)

Utworzenie indeksu przestrzennego:

Query | Query History

```
CREATE INDEX idx_porto_ndvi_rast_gist ON schema_minor.porto_ndvi
USING gist (ST_ConvexHull(rast));
```

Data Output | Messages | Notifications

```
CREATE INDEX

Query returned successfully in 69 msec.
```

Dodanie constraintów:

Query | Query History

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

Data Output | Messages | Notifications

	addrasterconstraints boolean
1	true

Przykład 2 - Funkcja zwrotna

Query Query History

```
create or replace function schema_minor.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;


CREATE TABLE schema_minor.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],
'schema_minor.ndvi(double precision[],
integer[],text[])'::regprocedure, --> This is the function!
'32BF'::text
) AS rast
FROM r;
```

Data Output Messages Notifications

```
SELECT 23

Query returned successfully in 169 msec.
```

Utworzenie indeksu przestrzennego i dodanie constraintów:

Query Query History

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

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

Data Output Messages Notifications

	addrasterconstraints boolean
1	true

Query Query History

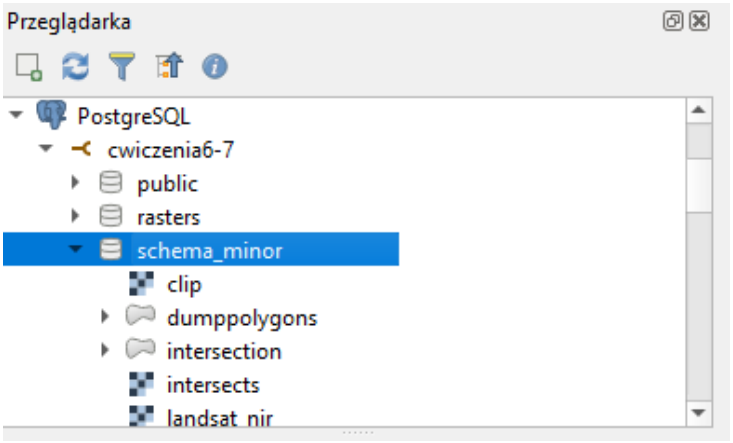
```
SELECT * FROM schema_minor.porto_ndvi2
```

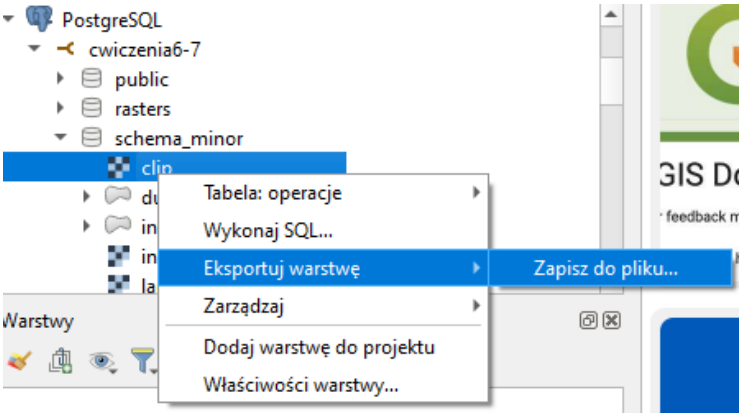
Data Output Messages Notifications

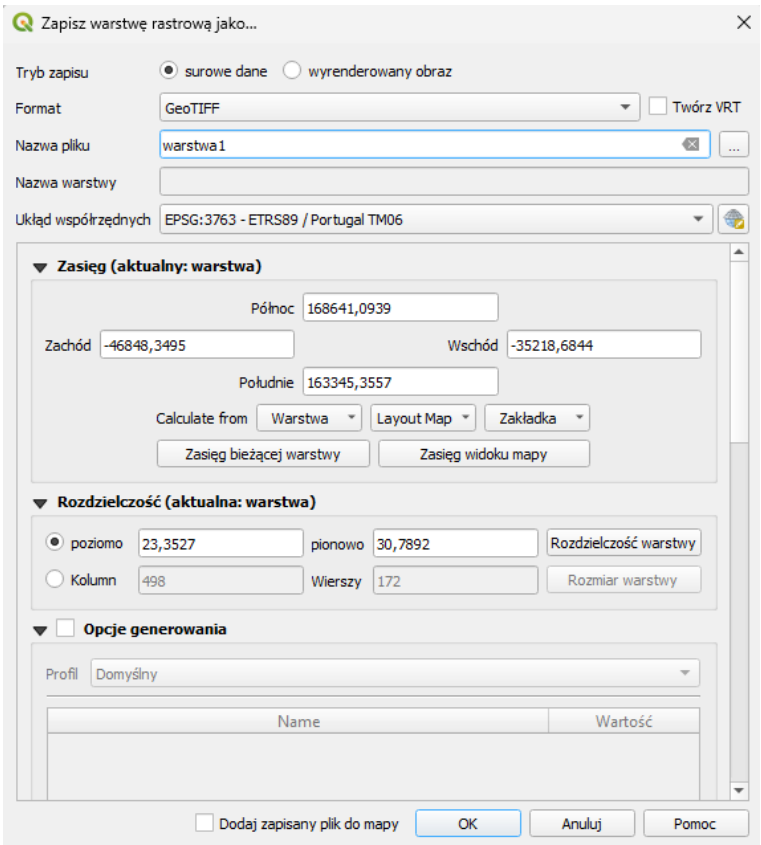
	rid integer	rast raster
1	245	0100000100384 9EE0BB84F3E404F2001E9AEB43DC0086F60B09B56E3C01F1195122C32044100000000000000000000000000000000B30E00000
2	270	0100000100384 9EE0BB84F3E404F2001E9AEB43DC075735DC2074FE3C0E4374673AE11044100000000000000000000000000000000B30E00003
3	246	0100000100384 9EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C075C249A27255044100000000000000000000000000000000B30E00002
4	270	0100000100384 9EE0BB84F3E404F2001E9AEB43DC03E4E2033AFB3E2C0E4374673AE11044100000000000000000000000000000000B30E00004
5	247	0100000100384 9EE0BB84F3E404F2001E9AEB43DC0AB0CA038D66DE1C079E13D89743B044100000000000000000000000000000000B30E0000

Eksport danych

Przykład 0 – QGIS







Przykład 1 - ST_AsTiff

```
SELECT ST_AsTiff(ST_Union(rast)) FROM schema_minor.porto_ndvi
```

	st_astiff bytea
1	[binary dat...

Przykład 2 - ST_AsGDALRaster

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

	st_asgdalraster bytea
1	[binary data]

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

SELECT ST_GDALDrivers()
```

	st_gdaldrivers record
1	0,GTiff,GeoTIFF,t,t,<CreationOptionList> <Option name='COMPRESS' type='string-select'> <Value>NONE</Value> <Value>LZW</Value> <Value>PACKBITS</Value> <Valu
2	1,AAIGrid,Arc/Info ASCII Grid,t,t,<CreationOptionList>
3	2,DTED,DTED Elevation Raster,t,t,
4	3,PNG,Portable Network Graphics,t,t,<CreationOptionList>
5	4,JPEG,JPEG JFIF,t,t,<CreationOptionList>
6	5,GIF,Graphics Interchange Format (.gif),t,t,<CreationOptionList>
7	6,USGSDEM,USGS Optional ASCII DEM (and CDED),t,t,<CreationOptionList> <Option name='PRODUCT' type='string-select' description='Specific Product Type'> <Value>DEFAULT</
8	7,XYZ,ASCII Gridded XYZ,t,t,<CreationOptionList> <Option name='COLUMN_SEPARATOR' type='string' default=' ' description='Separator between fields'/> <Option name='ADD_HEAD

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_minor.porto_ndvi;
----------------------------------------------
SELECT lo_export(loid, 'C:\new_folder\myraster.tiff') --> Save the file in a place where the user postgres have ac
FROM tmp_out;
----------------------------------------------
SELECT lo_unlink(loid)
FROM tmp_out; --> Delete the large object.
```

	lo_unlink integer
1	1