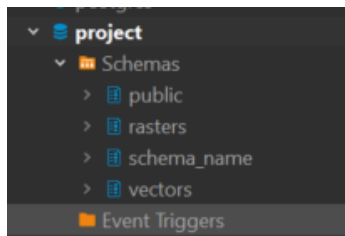


lab 6

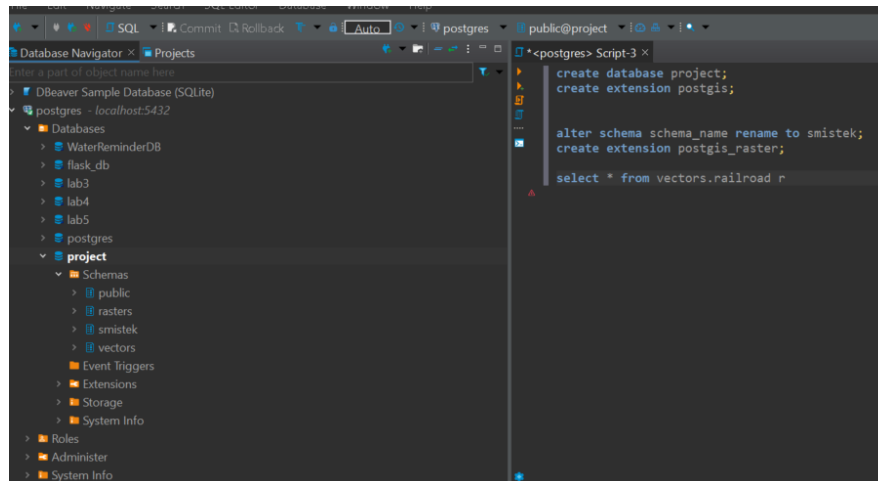
Olga ŚmisteK 400688 liAD

1. Restoring database from postgis_raster.backup file.

```
create database project;  
create extension postgis;
```



2. Renaming public schema.



3. Loading raster data.

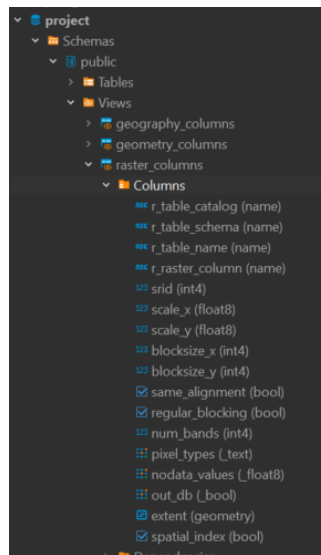
```
create extension postgis_raster;
```

```
C:\Program Files\PostgreSQL\15\bin>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d C:\Users\Olga\Desktop\SpatialDatabases\SpatialDatabases\lab6\srtm_1arc_v3.tif rasters.dem | psql -d project -h localhost -p 5432 -U postgres
Password for user postgres: Processing 1/1: C:\Users\Olga\Desktop\SpatialDatabases\SpatialDatabases\lab6\srtm_1arc_v3.tif

BEGIN
NOTICE: table "dem" does not exist, skipping
DROP TABLE
CREATE TABLE
```

```
C:\Program Files\PostgreSQL\15\bin>raster2pgsql.exe -s 3763 -N -32767 -t 128x128 -I -C -M -d C:\Users\Olga\Desktop\SpatialDatabases\SpatialDatabases\lab6\Landsat8_L1TP_RGBN.tif rasters.landsat8 | psql -d project -h localhost -U postgres -p 5432
Processing 1/1: C:\Users\Olga\Desktop\SpatialDatabases\SpatialDatabases\lab6\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
```



4. Creating rasters out of those already existing and interaction with vectors.

a. raster and vector intersection:

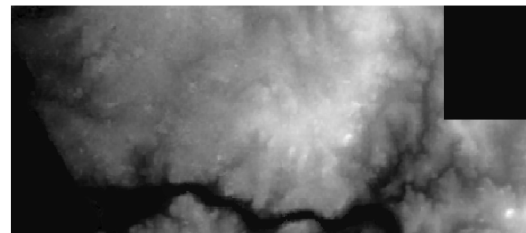
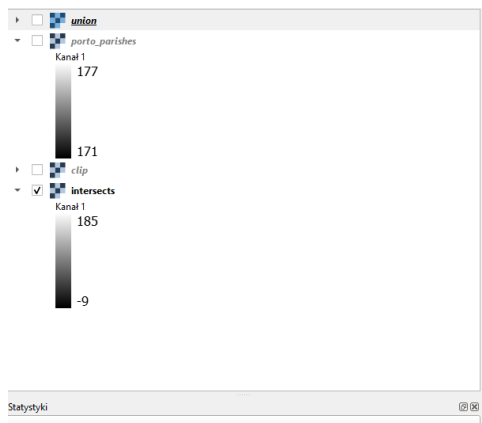
```
CREATE TABLE smistek.intersects AS
SELECT a.rast, b.municipality
FROM rasters.dem AS a, vectors.porto_parishes AS b
WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto';

alter table smistek.intersects
add column rid SERIAL PRIMARY KEY;

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

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

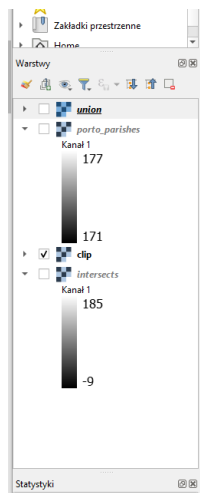
select * from smistek.intersects i;
```



b. raster clipping based on vector

```
CREATE TABLE smistek.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';

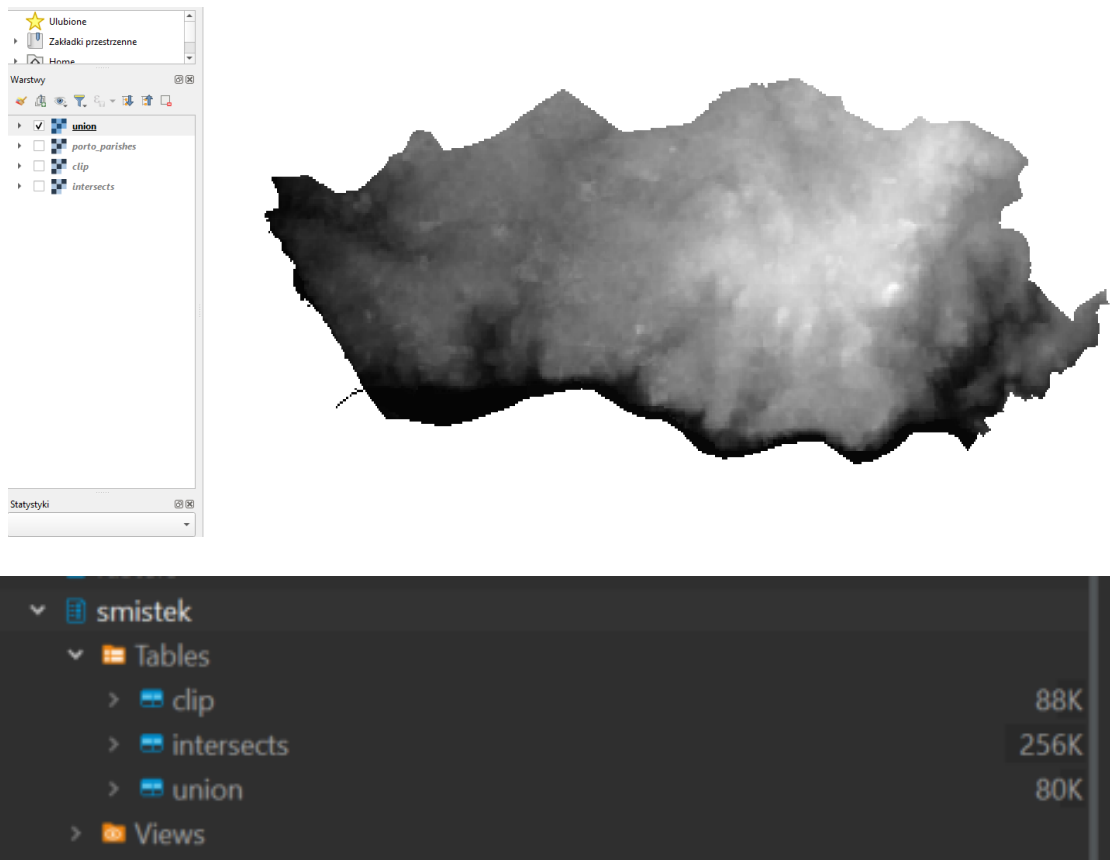
select * from smistek.clip;
```



c. combining multiple elements into one raster

```
CREATE TABLE smistek.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 smistek.union;
```

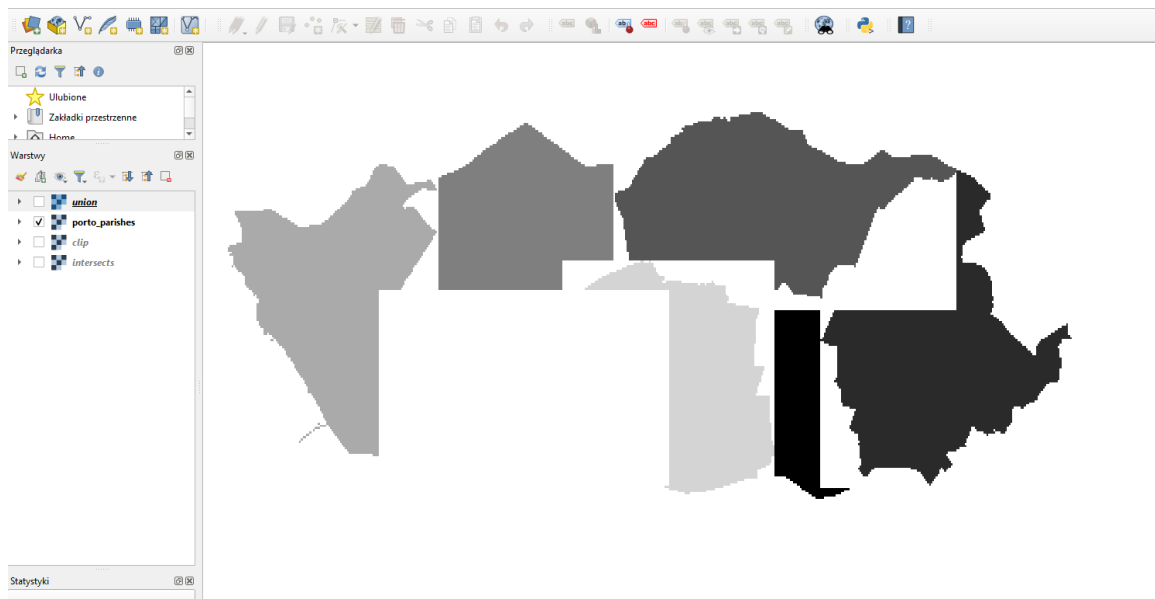


5. Rastering (creating rasters out of vectors)

a. ST_AsRaster

```
CREATE TABLE smistek.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';

select * from smistek.porto_parishes;
```



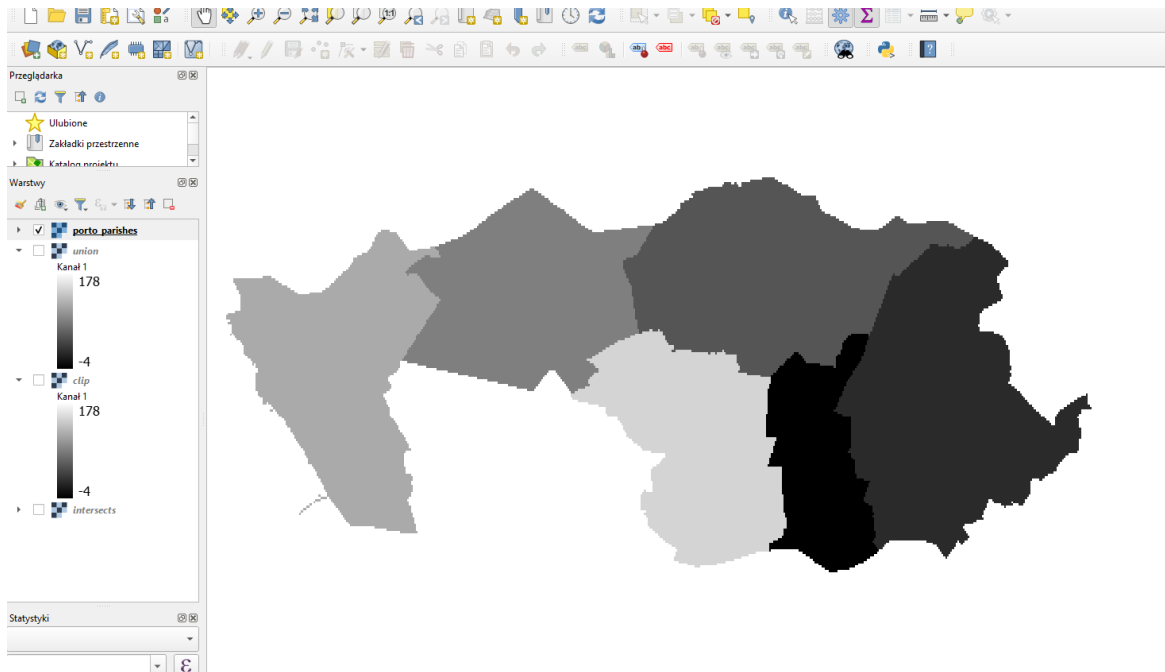
b. ST_Union

```

DROP TABLE smistek.porto_parishes; --> drop table porto_parishes first

CREATE TABLE smistek.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';

```



c. ST_Tile

```
DROP TABLE smistek.porto_parishes; --> drop table porto_parishes first
```

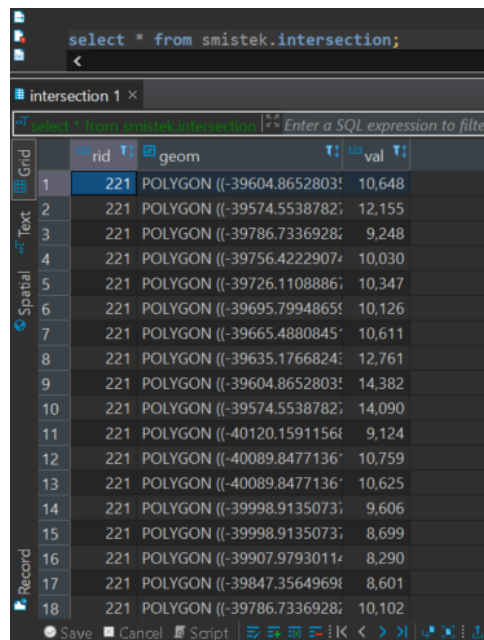
```
CREATE TABLE smistek.porto_parishes AS
WITH r AS (
  SELECT rast FROM rasters.dem
  LIMIT 1 )
SELECT st_tile(st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-
32767)),128,128,true,-32767) AS rast
FROM vectors.porto_parishes AS a, r
WHERE a.municipality ilike 'porto';
```



6. Converting rasters to vectors

a. ST_Intersection

```
create table smistek.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);
```



	rid	geom	val
1	221	POLYGON ((-39604.8652803;	10,648
2	221	POLYGON ((-39574.5538782;	12,155
3	221	POLYGON ((-39786.7336928;	9,248
4	221	POLYGON ((-39756.4222907;	10,030
5	221	POLYGON ((-39726.1108886;	10,347
6	221	POLYGON ((-39695.7994865;	10,126
7	221	POLYGON ((-39665.4880845;	10,611
8	221	POLYGON ((-39635.1766824;	12,761
9	221	POLYGON ((-39604.8652803;	14,382
10	221	POLYGON ((-39574.5538782;	14,090
11	221	POLYGON ((-40120.1591156;	9,124
12	221	POLYGON ((-40089.8477136;	10,759
13	221	POLYGON ((-40089.8477136;	10,625
14	221	POLYGON ((-39998.9135073;	9,606
15	221	POLYGON ((-39998.9135073;	8,699
16	221	POLYGON ((-39907.9793011;	8,290
17	221	POLYGON ((-39847.3564969;	8,601
18	221	POLYGON ((-39786.7336928;	10,102

b. ST_DumpAsPolygons

```
CREATE TABLE smistek.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);
```

select * from smistek.dumppolygons;

dumppolygons 1 x

	rid	geom	val
1	221	POLYGON ((-39665.4880845	12,761
2	221	POLYGON ((-39635.1766824	14,382
3	221	POLYGON ((-39604.8652803	14,090
4	221	POLYGON ((-40089.8477136	10,625
5	221	POLYGON ((-39907.9793011	8,601
6	221	POLYGON ((-39817.0450949	10,102
7	221	POLYGON ((-39786.7336928	11,050
8	221	POLYGON ((-39756.4222907	10,046
9	221	POLYGON ((-39726.1108886	9,908
10	221	POLYGON ((-39695.7994865	9,186
11	221	POLYGON ((-40150.4705177	10,427
12	221	POLYGON ((-40120.1591156	10,360
13	221	POLYGON ((-40089.8477136	10,139
14	221	POLYGON ((-40059.5363115	9,619
15	221	POLYGON ((-40029.2249094	9,606
16	221	POLYGON ((-39998.9135073	8,699
17	221	POLYGON ((-39847.3564969	9,102
18	221	POLYGON ((-39665.4880845	9,849

7. Rasters analysis.

a. ST_Band

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

select * from smistek.landsat_nir;

landsat_nir 1 x

	rid	rast
1	1	01000001003849EE0B884F3E404F2001E9AEB43DC02068193DB8C
2	2	01000001003849EE0B884F3E404F2001E9AEB43DC08C835ABC8FF
3	3	01000001003849EE0B884F3E404F2001E9AEB43DC0F99E9838C40
4	4	01000001003849EE0B884F3E404F2001E9AEB43DC066BADC8AC8
5	5	01000001003849EE0B884F3E404F2001E9AEB43DC0D2D51D3ACT
6	6	01000001003849EE0B884F3E404F2001E9AEB43DC03EF15EB9D15
7	7	01000001003849EE0B884F3E404F2001E9AEB43DC0A80CA038D6
8	8	01000001003849EE0B884F3E404F2001E9AEB43DC02F50C26FB51
9	9	01000001003849EE0B884F3E404F2001E9AEB43DC00887446EBE4
10	10	01000001003849EE0B884F3E404F2001E9AEB43DC0E0BDC66CC7
11	11	01000001003849EE0B884F3E404F2001E9AEB43DC08AF4486BD08
12	12	01000001003849EE0B884F3E404F2001E9AEB43DC0285796D382D
13	13	01000001003849EE0B884F3E404F2001E9AEB43DC0D8C49AD0C4
14	14	01000001003849EE0B884F3E404F2001E9AEB43DC088329CDD6
15	15	01000001003849EE0B884F3E404F2001E9AEB43DC078404795D12

b. ST_Clip

```
CREATE TABLE smistek.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);
```



```
select * from smistek.paranhos_dem;
```

rid	rast
380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2
382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6C7
412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE05
381	01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0DAE35DC0C

c. ST_Slope

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

```
select * from smistek.paranhos_slope;
```

rid	rast
380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2
382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6C7
412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE05
381	01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0DAE35DC0C

d. ST_Reclass

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

```
select * from smistek.paranhos_slope_reclass;
```

rid	st_reclass
380	01000001006172BF3E4D5A374080318D6907CA3EC09A49D3957D46E4C033B2707F2
382	01000001006172BF3E4D5A374080318D6907CA3EC02E3C8390DE87E2C0D7D06D6C7
412	01000001006172BF3E4D5A374080318D6907CA3EC0187635E2BF88E3C0474F11FE05
381	01000001006172BF3E4D5A374080318D6907CA3EC044951356C7ABE3C0DAE35DC0C

e. ST_SummaryStats

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

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

stats	count	sum	mean	stddev	min	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

f. ST_SummaryStats and Union

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

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

st_summarystats	count	sum	mean	stddev	min	max
1	9,977	1,222.455	122.5273128195	16.9080042027	87	158

g. ST_SummaryStats (with better control on complex data type)

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

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

min	max	mean
87	158	122.5273128195

h. ST_SummaryStats (with 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;
```

Enter a SQL expression to filter results (use Ctrl+Space)

	parish	min	max	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 Nevogi	-4	83	34.6673548979
6	União das freguesias de Cedofeita, Santo Ildefonso, Sé,	1	157	95.0027774104
7	União das freguesias de Lordelo do Ouro e Massarelos	-1	117	49.5005144033

i. ST_Value

```
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;
```

Enter a SQL expression to filter results (use Ctrl+Space)

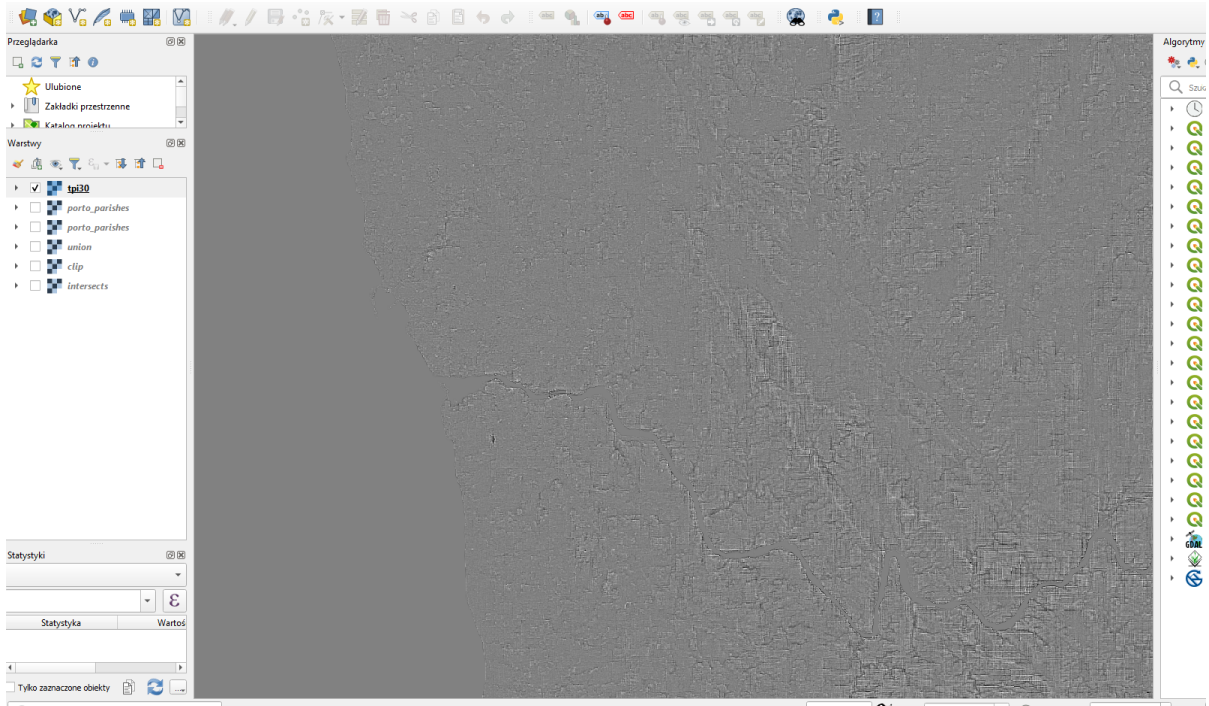
	name	st_value
1	Aldeia São Miguel	96
2	Alpendurada e Matos	145
3	Amarante	71
4	Baião	581
5	Cabeceiras de Basto	[NULL]
6	Castelo de Paiva	284
7	Celorico de Basto	227
8	Cinfães	405
9	Espinho	14
10	Fafe	338
11	Fajozes	53
12	Felgueiras	320
13	Gondomar	123
14	Guifões	69
15	Guimarães	197
16	Lousada	289
17	Maia	111
18	Marco de Canaveses	193

8. Topographic Position Index (TPI).

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

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

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

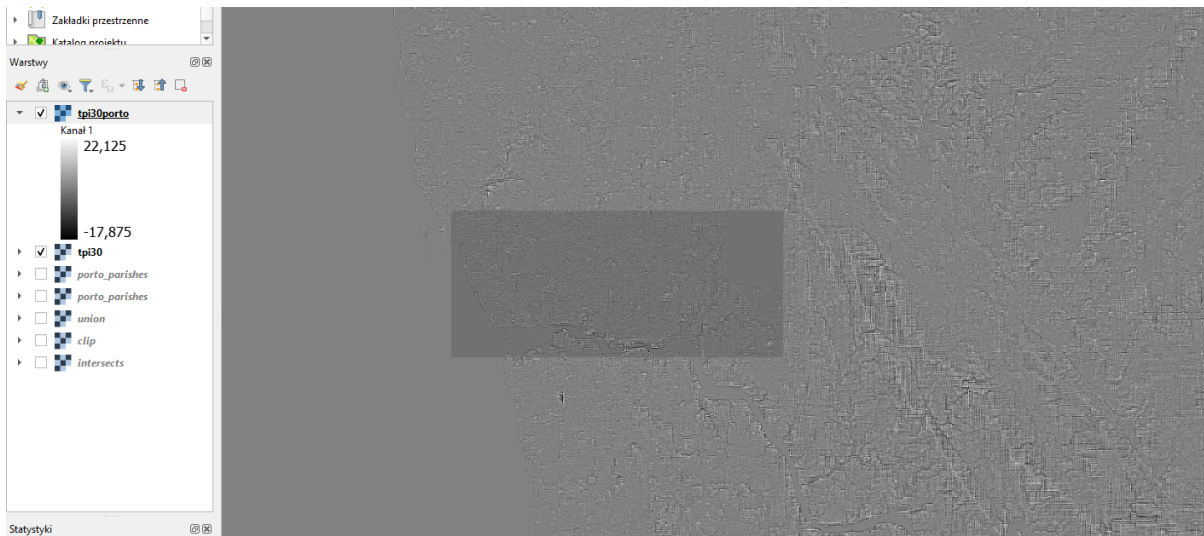


time: 36.11s

```
create table smistek.tpi30porto 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_rast_gist_porto ON smistek.tpi30porto
USING gist (ST_ConvexHull(rast));

SELECT AddRasterConstraints('smistek'::name,
'tpi30porto'::name,'rast'::name);
```



time: 1.701s

9. Map algebra.

NDVI=(NIR-Red)/(NIR+Red)

```
CREATE TABLE smistek.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;

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

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

select * from smistek.porto_ndvi;

porto_ndvi 1 x

select * from smistek.porto_ndvi | *Enter a SQL expression to filter results (use Ctrl+Space)*

	rid	rast
1	245	01000001003849EE0BB84F3E404F2001E9AEB43DC0086F60B09B56E3C01F1195122C3
2	270	01000001003849EE0BB84F3E404F2001E9AEB43DC075735DC2074FE3C0E4374673AE1
3	246	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C075C249A2725
4	270	01000001003849EE0BB84F3E404F2001E9AEB43DC03E4E2033AFB3E2C0E4374673AE1
5	247	01000001003849EE0BB84F3E404F2001E9AEB43DC0AB0CA038D66DE1C079E13D8974
6	246	01000001003849EE0BB84F3E404F2001E9AEB43DC075B133BEF0E4E2C00224CA124B7
7	221	01000001003849EE0BB84F3E404F2001E9AEB43DC0C069108BB315E5C086C41B00DC
8	245	01000001003849EE0BB84F3E404F2001E9AEB43DC0D2D51D3ACD37E5C0653CEA2E8
9	244	01000001003849EE0BB84F3E404F2001E9AEB43DC08A8A5008A0B8E5C02644A7A801
10	244	01000001003849EE0BB84F3E404F2001E9AEB43DC0D4DDC44A29E0E6C014B4B85D2
11	268	01000001003849EE0BB84F3E404F2001E9AEB43DC0AF24956AEA66E6C0E4374673AE
12	221	01000001003849EE0BB84F3E404F2001E9AEB43DC0E31A995A96E63C0EC74E7E06D5
13	245	01000001003849EE0BB84F3E404F2001E9AEB43DC08863BE794F45E4C0653CEA2E818
14	246	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0653CEA2E818
15	222	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0EC74E7E06D9
16	245	01000001003849EE0BB84F3E404F2001E9AEB43DC09BB0E0AA749CE4C07E0AC14760
17	269	01000001003849EE0BB84F3E404F2001E9AEB43DC008F3B3B8C92AE4C0E4374673AE1
18	270	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0E4374673AE1

```

create or replace function smistek.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 smistek.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],
'smistek.ndvi(double precision[],
integer[],text[])::regprocedure, --> This is the function!
'32BF'::text
) AS rast
FROM r;

```

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

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

select * from smistek.porto_ndvi2;

porto_ndvi2 1 x

select * from smistek.porto_ndvi2 Enter a SQL expression to filter results (use Ctrl+Space)

	rid	rast
1	245	01000001003849EE0BB84F3E404F2001E9AEB43DC0086F60B09B56E3C01F1195122C3
2	270	01000001003849EE0BB84F3E404F2001E9AEB43DC075735DC2074FE3C0E4374673AE1
3	246	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C075C249A2725
4	270	01000001003849EE0BB84F3E404F2001E9AEB43DC03E4E2033AFB3E2C0E4374673AE1
5	247	01000001003849EE0BB84F3E404F2001E9AEB43DC0A80CA038D66DE1C079E13D8974
6	246	01000001003849EE0BB84F3E404F2001E9AEB43DC075B133BEF0E4E2C00224CA124B7
7	221	01000001003849EE0BB84F3E404F2001E9AEB43DC0C069108BB315E5C0B6C41B00DC9
8	245	01000001003849EE0BB84F3E404F2001E9AEB43DC0D2D51D3ACD37E5C0653CEA2E8
9	244	01000001003849EE0BB84F3E404F2001E9AEB43DC08A8A5008A0B8E5C02644A7A801
10	244	01000001003849EE0BB84F3E404F2001E9AEB43DC0D4DDC44A29E0E6C014B4B85D2E
11	268	01000001003849EE0BB84F3E404F2001E9AEB43DC0AF24956AEA66E6C0E4374673AE1
12	221	01000001003849EE0BB84F3E404F2001E9AEB43DC0E31A995A96E6E3C0EC74E7E06D5
13	245	01000001003849EE0BB84F3E404F2001E9AEB43DC08863BE794F45E4C0653CEA2E818
14	246	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0653CEA2E818
15	222	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0EC74E7E06D9
16	245	01000001003849EE0BB84F3E404F2001E9AEB43DC09B80E0AA749CE4C07E0AC14760
17	269	01000001003849EE0BB84F3E404F2001E9AEB43DC008F3B3B8C92AE4C0E4374673AE1
18	270	01000001003849EE0BB84F3E404F2001E9AEB43DC03EF15EB9D152E3C0E4374673AE1

10. Data export.


```
--data export
SELECT ST_AsTiff(ST_Union(rast))
FROM smistek.porto_ndvi;
```

Adding pixel type
Adding nodata val
Adding out-of-dat
Adding maximum ex

Results 1 x

ST Select ST_AsTiff (smistek.porto_ndvi) Enter a SQL expression to filter results (use Ctrl+Space)

Grid

1 II* 3 ... [275539]

Value x

	00	01	02	03	04	05	06	07	08	09	0A
00000:49	49	2A	00	08	00	00	00	10	00	00	II*
0000B:01	03	00	01	00	00	00	80	01	00	00	;
00016:01	01	03	00	01	00	00	00	B3	00	00	;
00021:00	02	01	03	00	01	00	00	00	20	00	;
0002C:00	00	03	01	03	00	01	00	00	00	01	;
00037:00	00	00	06	01	03	00	01	00	00	00	;
00042:01	00	00	00	11	01	04	00	24	00	00	;
0004D:00	16	01	00	00	15	01	03	00	01	00	;
00058:00	00	01	00	00	00	16	01	03	00	01	;
00063:00	00	00	05	00	00	00	17	01	03	00	;
0006E:24	00	00	00	0E	00	00	00	1C	01	03	;
00079:00	01	00	00	00	01	00	00	53	01	00	;
00084:03	00	01	00	00	00	03	00	00	0E	00	;
0008F:83	0C	00	03	00	00	00	AC	01	00	00	;
0009A:82	84	0C	00	06	00	00	00	C4	01	00	;
000A5:00	AF	87	03	00	20	00	00	F4	01	00	;
000B0:00	00	B1	87	02	00	1F	00	00	00	34	;
000BB:02	00	00	81	A4	02	00	06	00	00	00	;
000C6:A6	01	00	00	00	00	00	00	1E	00	00	;
000D1:1E	00	1E	00	1E	00	1E	00	1E	00	1E	;
000DC:00	1E	00	1E	00	1E	00	1E	00	1E	00	;

Save Cancel Script 200 1 Rows: 1 1 row(s) fetched - 120ms (11ms fetch) on 2022-12-04

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

Adding nodata va
Adding out-of-da
Adding maximum e

Results 1 x

ST Select ST_AsGDALRaster (ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE', 'PREDICTOR=2', 'PZLEVEL=9']) Enter a SQL expression to filter results (use Ctrl+Space)

Grid

1 II* 3 ... [148838]

Value x

	00	01	02	03	04	05	06	07	08	09	0A	0B
00000:49	49	2A	00	08	00	00	00	11	00	00	01	II*
0000C:03	00	01	00	00	00	80	01	00	00	01	01	;
00018:03	00	01	00	00	00	B3	00	00	00	02	01	;
00024:03	00	01	00	00	00	20	00	00	00	03	01	;
00030:03	00	01	00	00	00	08	00	00	00	06	01	;
0003C:03	00	01	00	00	00	01	00	00	00	11	01	;
00048:04	00	24	00	00	00	6A	01	00	00	15	01	;
00054:03	00	01	00	00	00	01	00	00	00	16	01	;
00060:03	00	01	00	00	00	05	00	00	00	17	01	;
0006C:04	00	24	00	00	00	DA	00	00	00	1C	01	;
00078:03	00	01	00	00	00	01	00	00	00	3D	01	;
00084:03	00	01	00	00	00	02	00	00	00	53	01	;
00090:03	00	01	00	00	00	03	00	00	00	0E	83	;
0009C:0C	00	03	00	00	00	00	02	00	00	82	84	;
000A8:0C	00	06	00	00	00	18	02	00	00	AF	87	;
000B4:03	00	20	00	00	00	48	02	00	00	B1	87	;
000C0:02	00	1F	00	00	00	88	02	00	00	81	A4	;
000CC:02	00	06	00	00	00	FA	01	00	00	00	00	;
000D8:00	00	84	01	00	00	29	04	00	00	87	05	;
000E4:00	00	72	07	00	00	47	0E	00	00	74	0F	;
000F0:00	00	9F	10	00	00	2A	12	00	00	00	12	;

Save Cancel Script 200 1 Rows: 1 1 row(s) fetched - 57ms (2ms fetch) on 2022-12-04

```
SELECT ST_GDALDrivers();
```

Adding out-of-dat
Adding maximum e

Results 1 x

ST Select ST_GDALDrivers() Enter a SQL expression to filter results (use Ctrl+Space)

Grid

1 st_gdaldrivers


Value x

	(0,GTiff,GeoTIFF,t,t,<CreationOptionList>	<Option name='CON
1	(1,AAGrid,"Arc/Info ASCII Grid",t,t,<CreationOptionLi	
2	(2,DTED,"DTED Elevation Raster",t,t,"")	
3	(3,PNG,"Portable Network Graphics",t,t,<CreationOpti	
4	(4,JPEG,"JPEG JFIF",t,t,<CreationOptionList>f <Optic	
5	(5,GIF,"Graphics Interchange Format (.gif)",t,t,<Creati	
6	(6,USGSDEM,"USGS Optional ASCII DEM (and CDED)",	
7	(7,XYZ,"ASCII Gridded XYZ",t,t,<CreationOptionList>	


```

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 smistek.porto_ndvi;
-----
SELECT lo_export(loid,
'D:\myraster.tiff')
FROM tmp_out;
-----
SELECT lo_unlink(loid)
FROM tmp_out;

```

 myraster	04.12.2022 17:22	Obraz TIFF	146 KB
--	------------------	------------	--------