```
C:\Program Files\PostgreSQL\14\bin>pg_restore -h localhost -p 5432 -U postgres -d zad6 -v "D:\postgis_raster.backup"
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
pg_restore: warning: restoring tables WITH OIDS is not supported anymore
pg_restore: connecting to database for restore
Password:
pg_restore: creating SCHEMA "public"
pg_restore: while PROCESSING TOC:
pg_restore: from TOC entry 4; 2615 2200 SCHEMA public postgres
pg_restore: error: could not execute query: ERROR:  schema "public" already exists
Command was: CREATE SCHEMA public;


pg_restore: creating COMMENT "SCHEMA public"
pg_restore: creating SCHEMA "rasters"
pg_restore: creating SCHEMA "schema_name"
pg_restore: creating SCHEMA "vectors"
pg_restore: creating EXTENSION "plpgsql"
pg_restore: creating COMMENT "EXTENSION plpgsql"
pg_restore: creating EXTENSION "postgis"
pg_restore: creating COMMENT "EXTENSION postgis"
pg_restore: creating FUNCTION "public.bboxgeometryfrommetadata(character varying)"
pg_restore: creating FUNCTION "public.getisometadata(character varying, character varying)"
pg_restore: creating FUNCTION "public.registerisometadata(character varying, character varying, character varying)"
pg_restore: creating FUNCTION "public.update_imr_timestamp_column()"
pg_restore: creating TABLE "public.iso_metadata"
pg_restore: creating SEQUENCE "public.iso_metadata_id_seq"
pg_restore: creating SEQUENCE OWNED BY "public.iso_metadata_id_seq"
pg_restore: creating TABLE "public.iso_metadata_reference"
pg_restore: creating TABLE "vectors.porto_parishes"
pg_restore: creating SEQUENCE "vectors.Porto_Freguesias_id_seq"
pg_restore: creating SEQUENCE OWNED BY "vectors.Porto_Freguesias_id_seq"
pg_restore: creating TABLE "vectors.places"
pg_restore: creating TABLE "vectors.railroad"
pg_restore: creating DEFAULT "public.iso_metadata id"
pg_restore: creating DEFAULT "vectors.porto_parishes id"
pg_restore: processing data for table "public.iso_metadata"
pg_restore: processing data for table "public.iso_metadata_reference"
pg_restore: processing data for table "public.spatial_ref_sys"
pg_restore: processing data for table "vectors.places"
pg_restore: processing data for table "vectors.porto_parishes"
pg_restore: processing data for table "vectors.railroad"
pg_restore: executing SEQUENCE SET iso_metadata_id_seq
pg_restore: executing SEQUENCE SET Porto_Freguesias_id_seq
pg_restore: creating CONSTRAINT "public.iso_metadata iso_metadata_pkey"
pg_restore: creating CONSTRAINT "vectors.porto_parishes Porto_Freguesias_pkey"
pg_restore: creating CONSTRAINT "vectors.railroad ferrovia_pkey"
pg_restore: creating CONSTRAINT "vectors.places lugares_pkey"
pg_restore: creating INDEX "vectors.sidx_Porto_Freguesias_geom"
pg_restore: creating INDEX "vectors.sidx_ferrovia_geom"
pg_restore: creating INDEX "vectors.sidx_lugares_geom"
pg_restore: creating TRIGGER "public.iso_metadata_reference update_imr_timestamp"
pg_restore: creating FK CONSTRAINT "public.iso_metadata_reference iso_metadata_reference_md_file_id_fkey"
pg_restore: creating FK CONSTRAINT "public.iso_metadata_reference iso_metadata_reference_md_parent_id_fkey"
pg_restore: warning: errors ignored on restore: 1
```

```
C:\Program Files\PostgreSQL\14\bin>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d D:\srtm_1arc_v3.tif rasters
.dem > D:\BDP\dem.sql
Processing 1/1: D:\srtm_1arc_v3.tif
```

```
C:\Program Files\PostgreSQL\14\bin>raster2pgsql.exe -s 3763 -N -32767 -t 100x100 -I -C -M -d D:\srtm_1arc_v3.tif rasters
.dem | psql -d zad6 -h localhost -U postgres -p 5432
Processing 1/1: D:\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
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
```

```
C:\Program Files\PostgreSQL\14\bin>
C:\Program Files\PostgreSQL\14\bin>raster2pgsql.exe -s 3763 -N -32767 -t 128x128 -I -C -M -d D:\Landsat8_L1TP_RGBN.TIF r
asters.landsat8 | psql -d zad6 -h localhost -U postgres -p 5432
Processing 1/1: D:\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
INSERT 0 1
INSERT 0 1
```

```
zad6=# CREATE TABLE schema_name.intersects AS
zad6-# SELECT a.rast, b.municipality
zad6-# FROM rasters.dem AS a, vectors.porto_parishes AS b
zad6-# WHERE ST_Intersects(a.rast, b.geom) AND b.municipality ilike 'porto';
SELECT 25
zad6=#
```

```
zad6=# alter table schema_name.intersects
zad6-# add column rid SERIAL PRIMARY KEY;
ALTER TABLE
zad6=# CREATE INDEX idx_intersects_rast_gist ON schema_name.intersects
zad6-# USING gist (ST_ConvexHull(rast));
CREATE INDEX
zad6=# -- schema::name table_name::name raster_column::name
zad6=# SELECT AddRasterConstraints('schema_name'::name,
zad6(# 'intersects'::name,'rast'::name);
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)
```

```
zad6=# CREATE TABLE schema_name.clip AS
zad6-# SELECT ST_Clip(a.rast, b.geom, true), b.municipality
zad6-# FROM rasters.dem AS a, vectors.porto_parishes AS b
zad6-# WHERE ST_Intersects(a.rast, b.geom) AND b.municipality like 'PORTO';
SELECT 25
zad6=#
```

```
zad6=# CREATE TABLE schema_name.union AS
zad6-# SELECT ST_Union(ST_Clip(a.rast, b.geom, true))
zad6-# FROM rasters.dem AS a, vectors.porto_parishes AS b
zad6-# WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast);
SELECT 1
```

```
zad6=# CREATE TABLE schema_name.porto_parishes AS
zad6-# WITH r AS (
zad6(# SELECT rast FROM rasters.dem
zad6(# LIMIT 1
zad6(# )
zad6-# SELECT ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767) AS rast
zad6-# FROM vectors.porto_parishes AS a, r
zad6-# WHERE a.municipality ilike 'porto';
SELECT 7
zad6=#
```

```
zad6=# DROP TABLE schema_name.porto_parishes; --> drop table porto_parishes first
DROP TABLE
zad6=# CREATE TABLE schema_name.porto_parishes AS
zad6-# WITH r AS (
zad6(# SELECT rast FROM rasters.dem
zad6(# LIMIT 1
zad6(# )
zad6-# SELECT st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-32767)) AS rast
zad6-# FROM vectors.porto_parishes AS a, r
zad6-# WHERE a.municipality ilike 'porto';
SELECT 1
zad6=#
```

```
zad6=# DROP TABLE schema_name.porto_parishes; --> drop table porto_parishes first
DROP TABLE
zad6=# CREATE TABLE schema_name.porto_parishes AS
zad6-# WITH r AS (
zad6(# SELECT rast FROM rasters.dem
zad6(# LIMIT 1 )
zad6-# SELECT st_tile(st_union(ST_AsRaster(a.geom,r.rast,'8BUI',a.id,-
zad6(# 32767)),128,128,true,-32767) AS rast
zad6-# FROM vectors.porto_parishes AS a, r
zad6-# WHERE a.municipality ilike 'porto';
SELECT 8
zad6=#
```

```
zad6=# create table schema_name.intersection as
zad6-# SELECT
zad6-# a.rid,(ST_Intersection(b.geom,a.rast)).geom,(ST_Intersection(b.geom,a.rast)
zad6(# ).val
zad6-# FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
zad6-# WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
SELECT 6629
zad6=#
zad6=#
```

```
zad6=# CREATE TABLE schema_name.dumppolygons AS
zad6-# SELECT
zad6-# a.rid,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).geom,(ST_DumpAsPolygons(ST_Clip(a.rast,b.geom))).val
zad6-# FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
zad6-# WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
SELECT 6422
```

```
zad6=# CREATE TABLE schema_name.landsat_nir AS
zad6-# SELECT rid, ST_Band(rast,4) AS rast
zad6-# FROM rasters.landsat8;
SELECT 384
zad6=#
```

```
zad6=# CREATE TABLE schema_name.paranhos_dem AS
zad6-# SELECT a.rid,ST_Clip(a.rast, b.geom,true) as rast
zad6-# FROM rasters.dem AS a, vectors.porto_parishes AS b
zad6-# WHERE b.parish ilike 'paranhos' and ST_Intersects(b.geom,a.rast);
SELECT 4
zad6=#
```

```
zad6=# CREATE TABLE schema_name.paranhos_slope AS
zad6-# SELECT a.rid,ST_Slope(a.rast,1,'32BF','PERCENTAGE') as rast
zad6-# FROM schema_name.paranhos_dem AS a;
SELECT 4
zad6=#
```

```
zad6=# CREATE TABLE schema_name.paranhos_slope_reclass AS
zad6-# SELECT a.rid,ST_Reclass(a.rast,1,']0-15]:1, (15-30]:2, (30-9999:3',
zad6(# '32BF',0)
zad6-# FROM schema_name.paranhos_slope AS a;
SELECT 4
zad6=#
```

```
zad6=# SELECT st_summarystats(a.rast) AS stats
zad6-# FROM schema_name.paranhos_dem AS a;
                          stats
---------------------------------------------------------------
 (2616,278385,106.41628440366972,11.622628762211638,87,143)
 (6463,816615,126.35231316725978,14.0438229209133,94,158)
 (682,95581,140.14809384164224,12.078072186605759,103,158)
 (216,31874,147.5648148148148,4.262830628315728,137,158)
(4 rows)
```

```
zad6=# SELECT st_summarystats(ST_Union(a.rast))
zad6-# FROM schema_name.paranhos_dem AS a;
                       st_summarystats
---------------------------------------------------------------
 (9977,1222455,122.52731281948482,16.908004202736272,87,158)
(1 row)
```

```
zad6=# WITH t AS (
zad6(# SELECT st_summarystats(ST_Union(a.rast)) AS stats
zad6(# FROM schema_name.paranhos_dem AS a
zad6(# )
zad6-# SELECT (stats).min,(stats).max,(stats).mean FROM t;
 min | max |        mean
-----+-----+--------------------
  87 | 158 | 122.52731281948482
(1 row)
```

```
zad6=# WITH t AS (
zad6(# SELECT b.parish AS parish, st_summarystats(ST_Union(ST_Clip(a.rast,
zad6(# b.geom,true))) AS stats
zad6(# FROM rasters.dem AS a, vectors.porto_parishes AS b
zad6(# WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
zad6(# group by b.parish
zad6(# )
zad6-# SELECT parish,(stats).min,(stats).max,(stats).mean FROM t;
                                    parish                                     | min | max |       mean
-------------------------------------------------------------------------------+-----+-----+-------------------
 Bonfim                                                                        |   1 | 159 |  107.5658842667906
 CampanhÒ                                                                      |   0 | 178 |  74.66732213085449
 Paranhos                                                                      |  87 | 158 | 122.52731281948482
 Ramalde                                                                       |  48 | 108 |  77.58444444444444
 UniÒo das freguesias de Aldoar, Foz do Douro e Nevogilde                      |  -4 |  83 |  34.66735489791237
 UniÒo das freguesias de Cedofeita, Santo Ildefonso, SÚ, Miragaia, SÒo Nicolau e Vit¾ria |   1 | 157 |  95.00277741039545
 UniÒo das freguesias de Lordelo do Ouro e Massarelos                          |  -1 | 117 |  49.50051440329218
(7 rows)
```

```
zad6=# SELECT b.name,st_value(a.rast,(ST_Dump(b.geom)).geom)
zad6-# FROM
zad6-# rasters.dem a, vectors.places AS b
zad6-# WHERE ST_Intersects(a.rast,b.geom)
zad6-# ORDER BY b.name;
          name           | st_value
-------------------------+----------
 Aldeia SÒo Miguel       |       96
 Alpendurada e Matos     |      145
 Amarante                |       71
 BaiÒo                   |      581
 Cabeceiras de Basto     |
 Castelo de Paiva        |      284
 Celorico de Basto       |      227
 CinfÒes                 |      405
 Espinho                 |       14
 Fafe                    |      338
 Fajozes                 |       53
 Felgueiras              |      320
 Gondomar                |      123
 Guif§es                 |       69
 GuimarÒes               |      197
 Lousada                 |      289
 Maia                    |      111
 Marco de Canaveses      |      193
 Matosinhos              |       29
 Paþos de Ferreira       |      300
 Paredes                 |      178
 Penafiel                |      281
 Porto                   |       81
 P¾voa de Varzim         |       15
 Rio do Moinhos          |      106
 SÒo Mamede de Infesta   |       97
 TorrÒo                  |       89
 Trofa                   |       32
 Valongo                 |      139
 Vila do Conde           |       12
 Vila Nova de FamalicÒo  |      116
 Vila Nova de Gaia       |       82
 Vizela                  |      156
(33 rows)
```

```
zad6=# create table schema_name.tpi30 as
zad6-# select ST_TPI(a.rast,1) as rast
zad6-# from rasters.dem a;
SELECT 589
zad6=#
```

```
zad6=# CREATE INDEX idx_tpi30_rast_gist ON schema_name.tpi30
zad6-# USING gist (ST_ConvexHull(rast));
CREATE INDEX
zad6=#
```

```
zad6=# CREATE INDEX idx_tpi30_rast_gist ON schema_name.tpi30
zad6-# USING gist (ST_ConvexHull(rast));
CREATE INDEX
zad6=# SELECT AddRasterConstraints('schema_name'::name,
zad6(# 'tpi30'::name,'rast'::name);
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)
```

```
zad6=# CREATE TABLE schema_name.porto_ndvi AS
zad6-# WITH r AS (
zad6(# SELECT a.rid,ST_Clip(a.rast, b.geom,true) AS rast
zad6(# FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
zad6(# WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
zad6(# )
zad6-# SELECT
zad6-# r.rid,ST_MapAlgebra(
zad6(# r.rast, 1,
zad6(# r.rast, 4,
zad6(# '([rast2.val] - [rast1.val]) / ([rast2.val] +
zad6'# [rast1.val])::float','32BF'
zad6(# ) AS rast
zad6-# FROM r;
SELECT 23
zad6=#
```

```
zad6=# CREATE INDEX idx_porto_ndvi_rast_gist ON schema_name.porto_ndvi
zad6-# USING gist (ST_ConvexHull(rast));
CREATE INDEX
zad6=#
```

```
zad6=# SELECT AddRasterConstraints('schema_name'::name,
zad6(# 'porto_ndvi'::name,'rast'::name);
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)
```

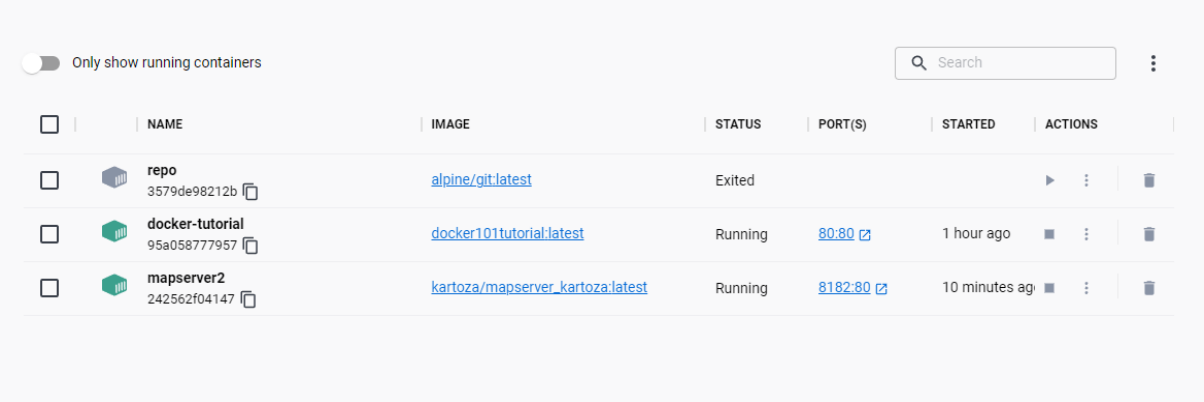
```
zad6=# create or replace function schema_name.ndvi(
zad6(# value double precision [] [] [],
zad6(# pos integer [][],
zad6(# VARIADIC userargs text []
zad6(# )
zad6-# RETURNS double precision AS
zad6-# $$
zad6$# BEGIN
zad6$# --RAISE NOTICE 'Pixel Value: %', value [1][1][1];-->For debug purposes
zad6$# RETURN (value [2][1][1] - value [1][1][1])/(value [2][1][1]+value
zad6$# [1][1][1]); --> NDVI calculation!
zad6$# END;
zad6$# $$
zad6-# LANGUAGE 'plpgsql' IMMUTABLE COST 1000;
CREATE FUNCTION
zad6=#
```

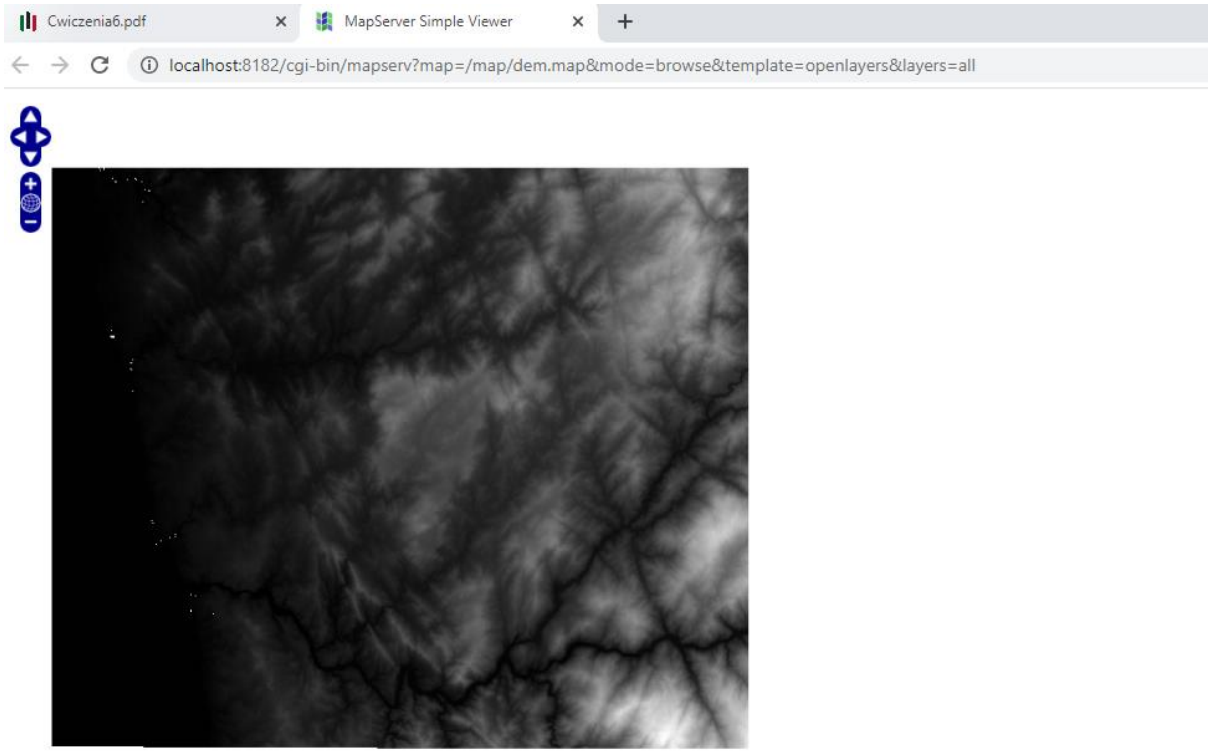
```
zad6=# CREATE TABLE schema_name.porto_ndvi2 AS
zad6-# WITH r AS (
zad6(# SELECT a.rid,ST_Clip(a.rast, b.geom,true) AS rast
zad6(# FROM rasters.landsat8 AS a, vectors.porto_parishes AS b
zad6(# WHERE b.municipality ilike 'porto' and ST_Intersects(b.geom,a.rast)
zad6(# )
zad6-# SELECT
zad6-# r.rid,ST_MapAlgebra(
zad6(# r.rast, ARRAY[1,4],
zad6(# 'schema_name.ndvi(double precision[],
zad6'# integer[],text[])'::regprocedure, --> This is the function!
zad6(# '32BF'::text
zad6(# ) AS rast
zad6-# FROM r;
SELECT 23
```

```
zad6=# CREATE INDEX idx_porto_ndvi2_rast_gist ON schema_name.porto_ndvi2
zad6-# USING gist (ST_ConvexHull(rast));
CREATE INDEX
zad6=#
```

```
zad6=# SELECT AddRasterConstraints('schema_name'::name,
zad6(# 'porto_ndvi2'::name,'rast'::name);
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)
```

```
zad6=# CREATE TABLE tmp_out AS
zad6-# SELECT lo_from_bytea(0,
zad6(#  ST_AsGDALRaster(ST_Union(rast), 'GTiff', ARRAY['COMPRESS=DEFLATE',
zad6(# 'PREDICTOR=2', 'PZLEVEL=9'])
zad6(#  ) AS loid
zad6-# FROM schema_name.porto_ndvi;
SELECT 1
zad6=# ----------------------------------------------
zad6=# SELECT lo_export(loid, 'G:\myraster.tiff') --> Save the file in a place
zad6-# where the user postgres have access. In windows a flash drive usualy works
zad6-# fine.
zad6-#  FROM tmp_out;
ERROR:  syntax error at or near "user"
LINE 2: where the user postgres have access. In windows a flash driv...
                 ^
zad6=# ----------------------------------------------
zad6=# SELECT lo_unlink(loid)
zad6-#  FROM tmp_out; --> Delete the large object.
 lo_unlink
-----------
         1
(1 row)
```




Cwiczenia6.pdf
MapServer Simple Viewer
localhost:8182/cgi-bin/mapserv?map=/map/dem.map&mode=browse&template=openlayers&layers=all