

# Ćwiczenia 7

## Zad. 1

```
.\raster2pgsql -s 4277 -N -32767 -t 100x100 -l -C -M -d  
C:\Users\Home\Desktop\7sem\SpatialDatabases\spatial-  
databases\cwiczenia7\ras250_gb\data\*.tif raster.uk_250k | psql -d cwi7 -h localhost -U  
postgres -p 5433
```

## Zad. 2 i 3

The screenshot shows a PostgreSQL command window with the following SQL commands:

```
CREATE DATABASE cwi7;  
CREATE EXTENSION postgis;  
CREATE EXTENSION postgis_raster;  
CREATE SCHEMA raster;  
-- .\raster2pgsql -s 4277 -N -32767 -t 100x100 -l -C -M -d C:\Users\Home\Desktop\7sem\SpatialDatabases\spatial-databases\cwiczen-  
--  
CREATE INDEX idx_rast_gist ON raster.uk_250k  
USING gist (ST_ConvexHull(rast));  
SELECT AddRasterConstraints('raster':name,  
                             'uk_250k':name, 'rast':name);  
SELECT ST_Union(rast) FROM raster.uk_250k;
```

The window also shows a progress bar and buttons for "Execute query - 741s" and "Cancel".

The screenshot shows a PostgreSQL command window with the following SQL commands:

```
CREATE DATABASE cwi7;  
CREATE EXTENSION postgis;  
CREATE EXTENSION postgis_raster;  
CREATE SCHEMA raster;  
-- .\raster2pgsql -s 4277 -N -32767 -t 100x100 -l -C -M -d C:\Users\Home\Desktop\7sem\SpatialDatabases\spatial-databases\cwiczen-  
--  
CREATE INDEX idx_rast_gist ON raster.uk_250k  
USING gist (ST_ConvexHull(rast));  
SELECT AddRasterConstraints('raster':name,  
                             'uk_250k':name, 'rast':name);
```

The window shows a long list of constraints being added, including:

- Adding number of bands constraint
- The constraint "enforce\_num\_bands\_rast" already exists. To replace it
- Adding pixel type constraint
- The constraint "enforce\_pixel\_type\_rast" already exists. To replace it
- Adding nodata value constraint
- The constraint "enforce\_nodata\_value\_rast" already exists. To replace it
- Adding out-of-database constraint
- The constraint "enforce\_out\_db\_rast" already exists. To replace it
- Adding maximum extent constraint
- The constraint "enforce\_max\_extent\_rast" already exists. To replace it

The window also shows a progress bar and buttons for "Execute query - 741s" and "Cancel".

Zapytanie trwało prawie 39 minut.

```

DROP TABLE IF EXISTS tmp_out ;

CREATE TABLE tmp_out AS
SELECT lo_from_bytea(0,
    ST_AsGDALRaster(ST_Union(rast), 'GTiff')
) AS loid
FROM raster.uk_250k;

SELECT lo_export(loid, 'C:\bazy_przestrzenne\zadanie3.tif')
FROM tmp_out;

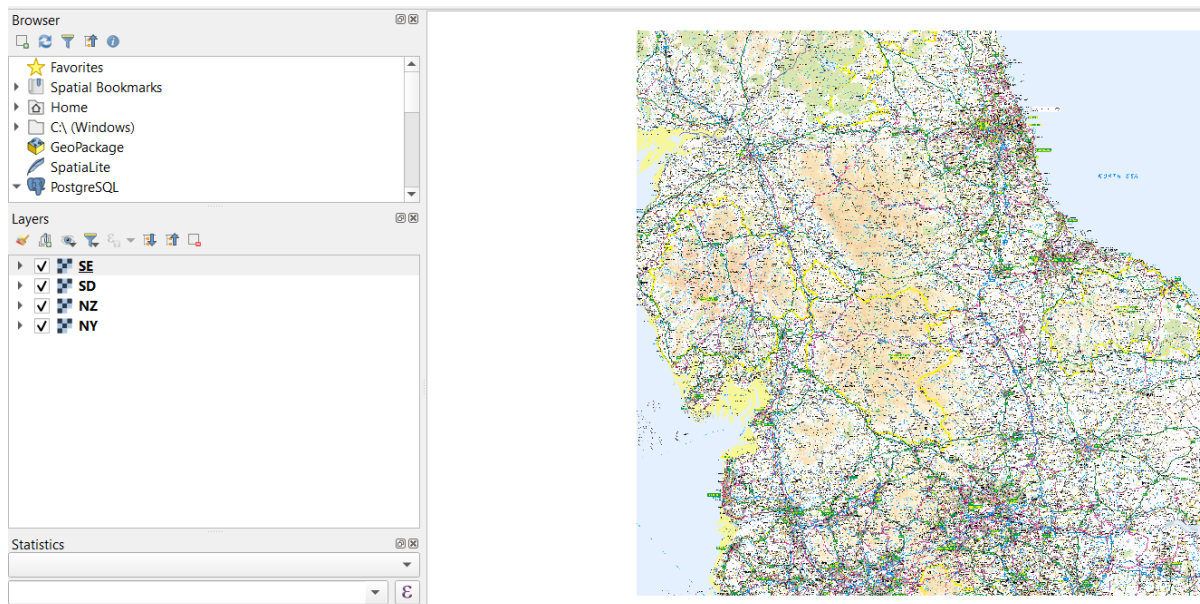
SELECT lo_unlink(loid)
FROM tmp_out;

```

Kwerenda również działała bardzo długo. Niestety nie udało mi się zapisać do pliku.

### Sposób z mniejszą ilością rastrow:

Spróbowałam wyciągnąć tylko zdjęcia, które zawierały w sobie obszar lake district. Były to 4 rastry:



Na tych 4 rastrach próbowałam zrobić poprzednie operacje:

```

.\raster2pgsql -s 4277 -N -32767 -t 100x100 -l -C -M -d
C:\Users\Home\Desktop\7sem\SpatialDatabases\spatial-databases\cwiczenia7\data\*.tif
uk_250k_4raster | psql -d cw7 -h localhost -U postgres -p 5433

```

```
-- ZAD 1-3 WITH smaller amount OF rasters
SELECT * FROM uk_250k_4raster;

CREATE INDEX idx_2_rast_gist ON uk_250k_4raster
USING gist(ST_ConvexHull(rast));

SELECT AddRasterConstraints('public'::name,
'uk_250k_4raster'::name, 'rast'::name);

CREATE TABLE tmp_out4r AS
SELECT lo_from_bytea(0,
ST_AsGDALRaster(ST_Union(rast), 'GTiff')
) AS loid
FROM uk_250k_4raster;

SELECT lo_export(loid, 'C:\bazy_przestrzenne\zadanie3.tif')
FROM tmp_out4r;

SELECT lo_unlink(loid)
FROM tmp_out4r;
```

Results 1

SELECT lo\_unlink(loid) FROM tmp\_out4r

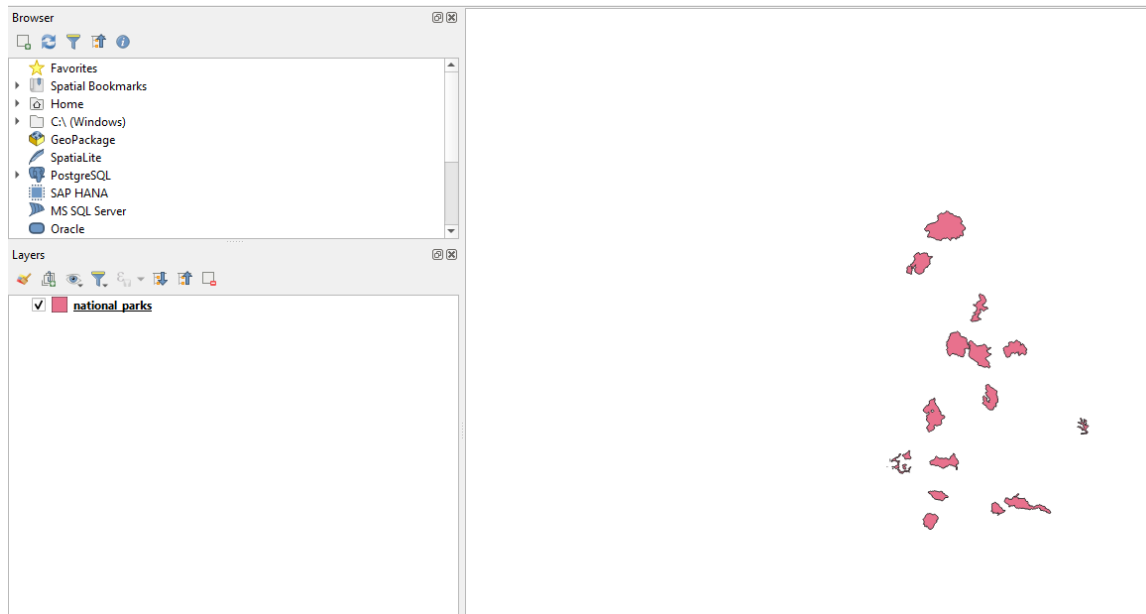
Grid	loid
1	1



## Zad. 4,5,6,7

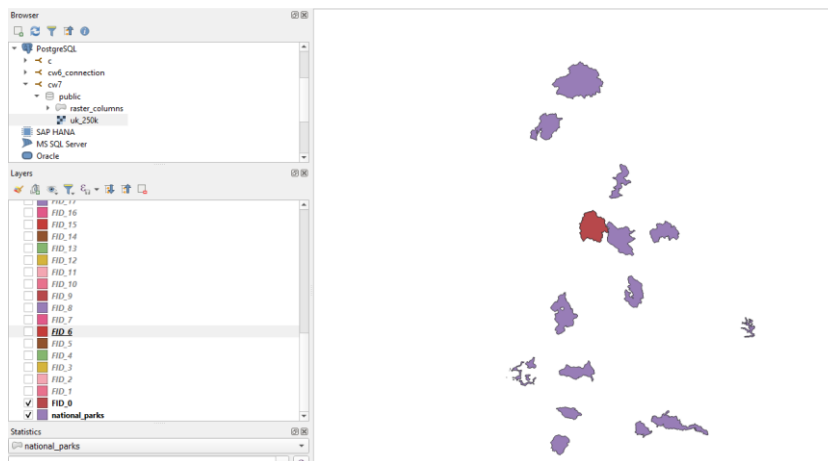
ogr2ogr.exe C:\bazy\_przestrzenne\ C:\bazy\_przestrzenne\OS\_Open\_Zoomstack.gpkg

Po wczytaniu do qgis:



```
shp2pgsql -s 27700 C:\bazy_przestrzenne\national_parks.shp national_parks | psql -U postgres -h localhost -p 5433 -d cwi7
```

gid	fid	geom
1	0	MULTIPOLYGON (((356612.4129999997 508769.34410000034, 356827.09399999996 508804.46900000005, 356900 508780, 357052.78100000004 508862.15599999995, 357279.46899999996 508984.03099999997, 357506.15199999998 509106.11199999999, 357732.82499999999 509228.19299999999, 357959.49899999999 509350.27399999999, 358186.17299999999 509472.35499999999, 358412.84699999999 509594.43599999999, 358639.52099999999 509716.51699999999, 358866.19499999999 509838.59799999999, 359092.86899999999 509960.67899999999, 359319.54299999999 510082.75999999999, 359546.21699999999 510204.84099999999, 359772.89099999999 510326.92199999999, 359999.56499999999 510448.00299999999, 360226.23899999999 510570.08399999999, 360452.91299999999 510692.16499999999, 360679.58699999999 510814.24599999999, 360906.26099999999 510936.32699999999, 361132.93499999999 511058.40799999999, 361359.60899999999 511180.48899999999, 361586.28299999999 511302.56999999999, 361812.95699999999 511424.65099999999, 362039.63099999999 511546.73199999999, 362266.30499999999 511668.81299999999, 362492.97899999999 511790.89399999999, 362719.65299999999 511912.97499999999, 362946.32699999999 512035.05599999999, 363173.00099999999 512157.13699999999, 363399.67499999999 512279.21799999999, 363626.34899999999 512401.29899999999, 363853.02299999999 512523.37999999999, 364079.69699999999 512645.46099999999, 364306.37099999999 512767.54199999999, 364533.04499999999 512889.62299999999, 364759.71899999999 513011.70399999999, 364986.39299999999 513133.78499999999, 365213.06699999999 513255.86599999999, 365439.74099999999 513377.94699999999, 365666.41499999999 513499.02799999999, 365893.08899999999 513621.10899999999, 366119.76299999999 513743.18999999999, 366346.43699999999 513865.27099999999, 366573.11099999999 513987.35199999999, 366799.78499999999 514109.43299999999, 367026.45899999999 514231.51399999999, 367253.13299999999 514353.59499999999, 367479.80699999999 514475.67599999999, 367706.48099999999 514597.75699999999, 367933.15499999999 514719.83799999999, 368159.82899999999 514841.91899999999, 368386.50299999999 514964.00099999999, 368613.17699999999 515086.08199999999, 368839.85099999999 515208.16299999999, 369066.52499999999 515330.24399999999, 369293.19899999999 515452.32499999999, 369519.87299999999 515574.40599999999, 369746.54699999999 515696.48699999999, 369973.22099999999 515818.56799999999, 370199.89499999999 515940.64899999999, 370426.56899999999 516062.72999999999, 370653.24299999999 516184.81099999999, 370879.91699999999 516306.89199999999, 371106.59099999999 516428.97299999999, 371333.26499999999 516551.05399999999, 371559.93899999999 516673.13499999999, 371786.61299999999 516795.21599999999, 372013.28699999999 516917.29699999999, 372240.00099999999 517039.37799999999, 372466.67499999999 517161.45899999999, 372693.34899999999 517283.53999999999, 372919.97299999999 517405.62099999999, 373146.64699999999 517527.70199999999, 373373.32099999999 517649.78299999999, 373599.99499999999 517771.86399999999, 373826.66899999999 517893.94499999999, 374053.34299999999 518016.02599999999, 374279.97699999999 518138.10699999999, 374506.65099999999 518260.18799999999, 374733.32499999999 518382.26899999999, 374959.99899999999 518504.34999999999, 375186.67299999999 518626.43099999999, 375413.34699999999 518748.51199999999, 375639.97099999999 518870.59299999999, 375866.64499999999 518992.67399999999, 376093.31899999999 519114.75499999999, 376319.99299999999 519236.83599999999, 376546.66699999999 519358.91699999999, 376773.34099999999 519481.00099999999, 376999.97499999999 519603.08199999999, 377226.64899999999 519725.16299999999, 377453.32299999999 519847.24399999999, 377679.99699999999 519969.32499999999, 377906.67099999999 520091.40599999999, 378133.34499999999 520213.48699999999, 378359.97899999999 520335.56799999999, 378586.65299999999 520457.64899999999, 378813.32699999999 520579.72999999999, 379039.97099999999 520701.81099999999, 379266.64499999999 520823.89199999999, 379493.31899999999 520945.97299999999, 379719.99299999999 521068.05399999999, 379946.66699999999 521190.13499999999, 380173.34099999999 521312.21599999999, 380399.97499999999 521434.29699999999, 380626.64899999999 521556.37799999999, 380853.32299999999 521678.45899999999, 381079.99699999999 521800.53999999999, 381306.67099999999 521922.62099999999, 381533.34499999999 522044.70199999999, 381759.97899999999 522166.78299999999, 381986.65299999999 522290.86399999999, 382213.32699999999 522412.94499999999, 382439.97099999999 522535.02599999999, 382666.64499999999 522657.10699999999, 382893.31899999999 522779.18799999999, 383119.97299999999 522901.26899999999, 383346.64699999999 523023.34999999999, 383573.32099999999 523145.43099999999, 383799.97499999999 523267.51199999999, 384026.64899999999 523389.59299999999, 384253.32299999999 523511.67399999999, 384479.97699999999 523633.75499999999, 384706.65099999999 523755.83599999999, 384933.32499999999 523877.91699999999, 385159.97899999999 524000.00099999999, 385386.65299999999 524122.08199999999, 385613.32699999999 524244.16299999999, 385839.97099999999 524366.24399999999, 386066.64499999999 524488.32499999999, 386293.31899999999 524610.40599999999, 386519.97299999999 524732.48699999999, 386746.64699999999 524854.56799999999, 386973.32099999999 524976.64899999999, 387199.97499999999 525098.72999999999, 387426.64899999999 525220.81099999999, 387653.32299999999 525342.89199999999, 387879.97699999999 525464.97299999999, 388106.65099999999 525587.05399999999, 388333.32499999999 525709.13499999999, 388559.97899999999 525831.21599999999, 388786.65299999999 525953.29699999999, 389013.32699999999 526075.37799999999, 389239.97099999999 526197.45899999999, 389466.64499999999 526319.53999999999, 389693.31899999999 526441.62099999999, 389919.97299999999 526563.70199999999, 390146.64699999999 526685.78299999999, 390373.32099999999 526807.86399999999, 390599.97499999999 526929.94499999999, 390826.64899999999 527052.02599999999, 391053.32299999999 527174.10699999999, 391279.97699999999 527296.18799999999, 391506.65099999999 527418.26899999999, 391733.32499999999 527540.34999999999, 391959.97899999999 527662.43099999999, 392186.65299999999 527784.51199999999, 392413.32699999999 527906.59299999999, 392639.97099999999 528028.67399999999, 392866.64499999999 528150.75499999999, 393093.31899999999 528272.83599999999, 393319.97299999999 528394.91699999999, 393546.64699999999 528516.99799999999, 393773.32099999999 528639.07899999999, 393999.97499999999 528761.15999999999, 394226.64899999999 528883.24099999999, 394453.32299999999 529005.32199999999, 394679.97699999999 529127.40299999999, 394906.65099999999 529249.48399999999, 395133.32499999999 529371.56499999999, 395359.97899999999 529493.64599999999, 395586.65299999999 529615.72699999999, 395813.32699999999 529737.80799999999, 396039.97099999999 529859.88899999999, 396266.64499999999 529981.96999999999, 396493.31899999999 530104.05099999999, 396719.97299999999 530226.13199999999, 396946.64699999999 530348.21299999999, 397173.32099999999 530470.29399999999, 397399.97499999999 530592.37499999999, 397626.64899999999 530714.45599999999, 397853.32299999999 530836.53699999999, 398079.97699999999 530958.61799999999, 398306.65099999999 531080.69899999999, 398533.32499999999 531202.77999999999, 398759.97899999999 531324.86099999999, 398986.65299999999 531446.94199999999, 399213.32699999999 531569.02299999999, 399439.97099999999 531691.10399999999, 399666.64499999999 531813.18499999999, 399893.31899999999 531935.26599999999, 399999.97299999999 532057.34699999999, 400000.00099999999 532179.42799999999, 400000.00099999999 532301.50899999999, 400000.00099999999 532423.58999999999, 400000.00099999999 532545.67099999999, 400000.00099999999 532667.75199999999, 400000.00099999999 532789.83299999999, 400000.00099999999 532911.91399999999, 400000.00099999999 533033.99499999999, 400000.00099999999 533156.07599999999, 400000.00099999999 533278.15699999999, 400000.00099999999 533400.23799999999, 400000.00099999999 533522.31899999999, 400000.00099999999 533644.39999999999, 400000.00099999999 533766.48099999999, 400000.00099999999 533888.56199999999, 400000.00099999999 534010.64299999999, 400000.00099999999 534132.72399999999, 400000.00099999999 534254.80499999999, 400000.00099999999 534376.88599999999, 400000.00099999999 534498.96699999999, 400000.00099999999 534621.04799999999, 400000.00099999999 534743.12899999999, 400000.00099999999 534865.20999999999, 400000.00099999999 534987.29099999999, 400000.00099999999 535109.37199999999, 400000.00099999999 535231.45299999999, 400000.00099999999 535353.53399999999, 400000.00099999999 535475.61499999999, 400000.00099999999 535597.69599999999, 400000.00099999999 535719.77699999999, 400000.00099999999 535841.85799999999, 400000.00099999999 535963.93899999999, 400000.00099999999 536086.01999999999, 400000.00099999999 536208.10099999999, 400000.00099999999 536330.18199999999, 400000.00099999999 536452.26299999999, 400000.00099999999 536574.34399999999, 400000.00099999999 536696.42499999999, 400000.00099999999 536818.50599999999, 400000.00099999999 536940.58699999999, 400000.00099999999 537062.66799999999, 400000.00099999999 537184.74899999999, 400000.00099999999 537306.82999999999, 400000.00099999999 537428.91099999999, 400000.00099999999 537550.99199999999, 400000.00099999999 537673.07299999999, 400000.00099999999 537795.15399999999, 400000.00099999999 537917.23499999999, 400000.00099999999 538039.31599999999, 400000.00099999999 538161.39699999999, 400000.00099999999 538283.47799999999, 400000.00099999999 538405.55899999999, 400000.00099999999 538527.63999999999, 400000.00099999999 538649.72099999999, 400000.00099999999 538771.80199999999, 400000.00099999999 538893.88299999999, 400000.00099999999 539015.96399999999, 400000.00099999999 539138.04499999999, 400000.00099999999 539260.12599999999, 400000.00099999999 539382.20699999999, 400000.00099999999 539504.28799999999, 400000.00099999999 539626.36899999999, 400000.00099999999 539748.44999999999, 400000.00099999999 539870.53099999999, 400000.00099999999 539992.61199999999, 400000.00099999999 540114.69299999999, 400000.00099999999 540236.77399999999, 400000.00099999999 540358.85499999999, 400000.00099999999 540480.93599999999, 400000.00099999999 540603.01699999999, 400000.00099999999 540725.09799999999, 400000.00099999999 540847.17899999999, 400000.00099999999 540969.25999999999, 400000.00099999999 541091.34099999999, 400000.00099999999 541213.42199999999, 400000.00099999999 541335.50299999999, 400000.00099999999 541457.58399999999, 400000.00099999999 541579.66499999999, 400000.00099999999 541701.74599999999, 400000.00099999999 541823.82699999999, 400000.00099999999 541945.90799999999, 400000.00099999999 542067.98899999999, 400000.00099999999 542190.06999999999, 400000.00099999999 542312.15099999999, 400000.00099999999 542434.23199999999, 400000.00099999999 542556.31299999999, 400000.00099999999 542678.39399999999, 400000.00099999999 542800.47499999999, 400000.00099999999 542922.55599999999, 400000.00099999999 543044.63699999999, 400000.00099999999 543166.71799999999, 400000.00099999999 543288.79899999999, 400000.00099999999 543410.87999999999, 400000.00099999999 543532.96099999999, 400000.00099999999 543655.04199999999, 400000.00099999999 543777.12299999999, 400000.00099999999 543899.20399999999, 400000.00099999999 544021.28499999999, 400000.00099999999 544143.36599999999, 400000.00099999999 544265.44699999999, 400000.00099999999 544387.52799999999, 400000.00099999999 544509.60899999999, 400000.00099999999 544631.68999999999, 400000.00099999999 544753.77099999999, 400000.00099999999 544875.85199999999, 400000.00099999999 544997.93299999999, 400000.00099999999 545120.01399999999, 400000.00099999999 545242.09499999999, 400000.00099999999 545364.17599999999, 400000.00099999999 545486.25699999999, 400000.00099999999 545608.33799999999, 400000



```
SELECT * FROM national_parks np;

SELECT UpdateGeometrySRID('national_parks','geom',4277);

CREATE TABLE uk_lake_district AS
SELECT a.rid,ST_Clip(a.rast, b.geom,true) as rast
FROM raster.uk_250k AS a, national_parks AS b
where b.gid = 1 and ST_Intersects(b.geom,a.rast);
```

```
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 uk_lake_district;

SELECT lo_export(lo_id, 'C:\bazy_przestrzenne\zadanie7.tif')
FROM tmp_out;

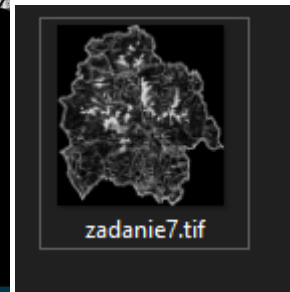
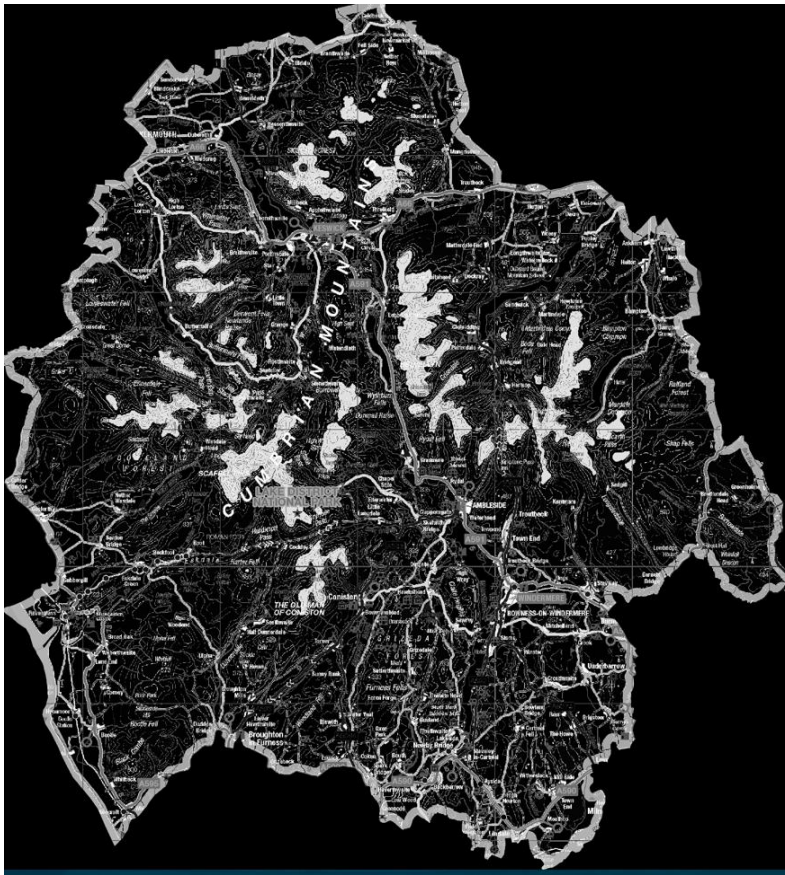
SELECT lo_unlink(lo_id)
FROM tmp_out; --> Delete the large object.
```

Results 1

SELECT lo\_unlink(lo\_id) FROM tmp\_out; Enter a SQL expression to filter results (use Ctrl+Space)

1	lo_unlink
1	





Zad. 8,9,10,11

Display 1 to 25 of 13678 products.

Order By: Ingestion Date

Request Done: ( footprint:Intersects(POLYGON((-1.7378849947673238 55.10638671217748,-1.7378849947673238 55.10638671217748,-4.234262054003056 55.06424970201394,-4.2432057253497435 54.04482450707297,-1.9252820930477539 55.06424970201394,-1.7378849947673238 55.10638671217748)))

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:45:01.024Z Size: 62.46 MB

☐ **MSI** S2B\_MSIL2A\_20221210T112349\_N0509\_R037\_T30UWE\_20221210T125828

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:23:49.024Z Size: 647.56 MB

☐ **MSI** S2B\_MSIL2A\_20221210T112349\_N0509\_R037\_T30UWG\_20221210T125828

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:23:49.024Z Size: 1.02 GB

☒ **MSI** S2B\_MSIL2A\_20221210T112349\_N0509\_R037\_T30UVF\_20221210T125828

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:23:49.024Z Size: 798.33 MB

☐ **MSI** S2B\_MSIL2A\_20221210T112349\_N0509\_R037\_T30UVE\_20221210T125828

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:23:49.024Z Size: 865.26 MB

☐ **MSI** S2B\_MSIL2A\_20221210T112349\_N0509\_R037\_T30UVQ\_20221210T125828

Download URL: <https://scihub.copernicus.eu/data/v1/Products/7147-6f1a-4d6e-09f-09f-1574954741>

Mission: Sentinel-2 Instrument: MSI Sensing Date: 2022-12-10T11:23:49.024Z Size: 562.65 MB

Products per page: 25 << page: 1 of 548 >>

```
raster2pgsql.exe -s 4277 -N -32767 -t 100x100 -l -C -M -d C:\bazy_przestrzenne\sentinel.jp2
raster.sentinel | psql -d cwif7 -h localhost -U postgres -p 5433
```



```

SELECT * FROM raster.sentinel;

CREATE INDEX idx_rast_sentinel_gist ON raster.sentinel
USING gist (ST_ConvexHull(rast));

SELECT AddRasterConstraints('raster'::name,
'sentinel'::name,'rast'::name);

CREATE OR REPLACE FUNCTION NDVI(
value double precision [] [] [],
pos integer [][],
VARIADIC userargs text []
)
RETURNS double precision AS
$$
BEGIN

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 NDVI_2 AS
WITH r AS (
SELECT * FROM raster.sentinel
)
SELECT
r.rid,ST_MapAlgebra(
r.rast, ARRAY[1,4],
'NDVI(double precision[],
integer[],text[])':regprocedure, --> This is the function!
'32BF'::text
) AS rast
FROM r;

SELECT * FROM NDVI_2;

```

```

SELECT * FROM NDVI_2;

CREATE TABLE uk_lake_district_sentinel AS
SELECT a.rid, ST_Clip(a.rast,b.geom,true) AS rast
FROM NDVI_2 AS a, national_parks AS b
WHERE b.gid=1 AND ST_Intersects(b.geom,a.rast);

SELECT * FROM uk_lake_district_sentinel;

DROP TABLE tmp_out2;

CREATE TABLE tmp_out2 AS
SELECT lo_from_bytea(0,
ST_AsGDALRaster(ST_Union(rast), 'Gtiff', ARRAY['COMPRESS=DEFLATE',
'PREDICTOR=2', 'PZLEVEL=0']))
) AS loid
FROM uk_lake_district_sentinel;

SELECT lo_export(loid, 'C:\bazy_przestrzenne\zadanie11.tiff')
FROM tmp_out2;

SELECT lo_unlink(loid)
FROM tmp_out2; --> Delete the large object.

```

Po przycięciu brak wyników, brak części wspólnej, mimo że mam te same układy współrzędnych.

```
SELECT DISTINCT ST_SRID(rast) FROM ndvi_2;
```

Results 1	
st_srid	
1	4,277

```
SELECT FIND_SRID('public','national_parks','geom');
```

Results 1	
find_srid	
1	4,277