

ÚVOD DO ZPRACOVÁNÍ PROSTOROVÝCH DAT

SEMESTRÁLNÍ PROJEKT

DOKUMENTACE

SKUPINA E

MICHAL KARÁSEK

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

KATEDRA GEOMATIKY

AKADEMICKÝ ROK 2017/2018, LETNÍ SEMESTR

Obsah

1 ÚVOD	3
1.1 LPIS (Land parcel identification system)	3
1.2 WFS (web feature service)	3
2 STAHOVÁNÍ DAT	4
2.1 Stahování dat.....	4
2.1.1 Náповěda	4
2.1.2 Spouštění programu	5
2.2 Tvorba dlaždic.....	7
2.2.1 Náповěda	7
2.2.2 Spouštění programu	8
3 VRSTVY	10
3.1 Výběr vrstev.....	10
3.2 Nahrání vrstev do postGIS.....	10
3.3 Odstranění duplicitních prvků.....	10
3.4 Vytvoření nové vrstvy	10
3.5 Vytvoření primárního klíče a prostorového indexu	11
4 PROSTOROVÉ DOTAZY.....	11
5 ZÁVĚR.....	14

1 ÚVOD

Cílem semestrálního projektu bylo vytvoření skriptu pro stahování datových vrstev z geoportálu LPIS. K vrstvám bylo přistupováno přes WFS webovou službu. Vrstvy byly následně nahrány do PostGIS databáze na serveru geol02.fsv.cvut.cz. Nad nahranými daty byla následně vytvořena sada prostorových dotazů.

1.1 LPIS (Land parcel identification system)

Geoportál spravovaný ministerstvem zemědělství, tvořený primárně evidencí využití zemědělské půdy. Primárně slouží k ověřování informací z žádostí o dotace poskytované ve vazbě na zemědělskou půdu, nehledě na zdroj finančních prostředků.

1.2 WFS (web feature service)

Jedná se o standard vyvinutý OGC (open geospatial consortium). Služba pracuje na principu server - klient a umožňuje sdílet data ve vektorové podobě. Data jsou vracena ve formátu GML.

2 STAHOVÁNÍ DAT

Výstupem práce jsou dva programy napsané ve skriptovacím programovacím jazyce python. Program pro stahování dat z LPIS, který přistupuje k serveru pomocí WFS služby a ukládající stažená data do gml souboru a dále program pro tvorbu pravidelné polygonové sítě pokrývající území ČR. Na základě známého id dlaždice v polygonové síti tak umožňuje stahovat pouze data pro konkrétní zájmové území.

2.1 Stahování dat

LPIS omezuje data poskytovaná přes WFS službu na maximální počet 200 prvků. Proto je požadované ohraničující území rozděleno do čtvercových dlaždic s definovanou délkou strany uživatelem. Ve skriptu je pak přistupováno k jednotlivým dlaždicím, které jsou testovány, zda obsahují data. Pokud ano, jsou uložena do gml souboru. K serveru je přistupováno přes knihovnu OWSLib 0.16.0.

2.1.1 Nápověda

Program je nutno spouštět v python3. S nižšími verzemi jazyka není kompatibilní. Nápovědu k povinným a volitelným argumentům při spouštění programu zobrazíme následujícím způsobem.

```
$python3 LPIS.py -h
```

Po zadání příkazu vrací program uživateli následující informace.

```
usage: LPIS.py [-h] [--Bbox BBOX [BBOX ...]] [--layers]
              [--Tiles TILES [TILES ...]]
              layerName tileSize

Script for downloading wfs layers from LPIS

positional arguments:
  layerName             Name of layer to download data from
  tileSize              Length of tile size [m]

optional arguments:
  -h, --help            show this help message and exit
  --Bbox BBOX [BBOX ...]
                        Bounding area coordinates. (default: (-904580,
                        -1229235, -430007, -934219))
  --layers              Type to get layers (default: False)
  --Tiles TILES [TILES ...]
                        Enter numbers of tiles from Tile.py. Tile size must be
                        set on same size. (default: None)
```

Povinné údaje:

layername ... typ string se jménem datové vrstvy. Program přijímá na vstupu pouze jednu vrstvu.

Tilesize ... typ integer. Velikost čtvercové dlaždice na které bude zájmové území děleno. Hodnota udávána v metrech.

Volitelné údaje:

--Bbox ... typ integer. Ohraničující oblast, pro kterou budou data stažena. Přednastaven ohraničující obdelník ČR.

--layers ... příkaz vypisující všechny vrstvy dostupné přes wfs službu

--Tiles ... typ integer. Číslo dlaždic, pro které chceme stáhnout data.

Poznámka: Pro vstup byla použita knihovna argparse pro kterou není s užitím jejích funkcí možné ošetřit závislosti povinných a nepovinných údajů vstupujících do programu. Pro zobrazení vrstev geoportálu LPIS použijte následující příkaz.

\$python3 LPIS.py x 0 --layers

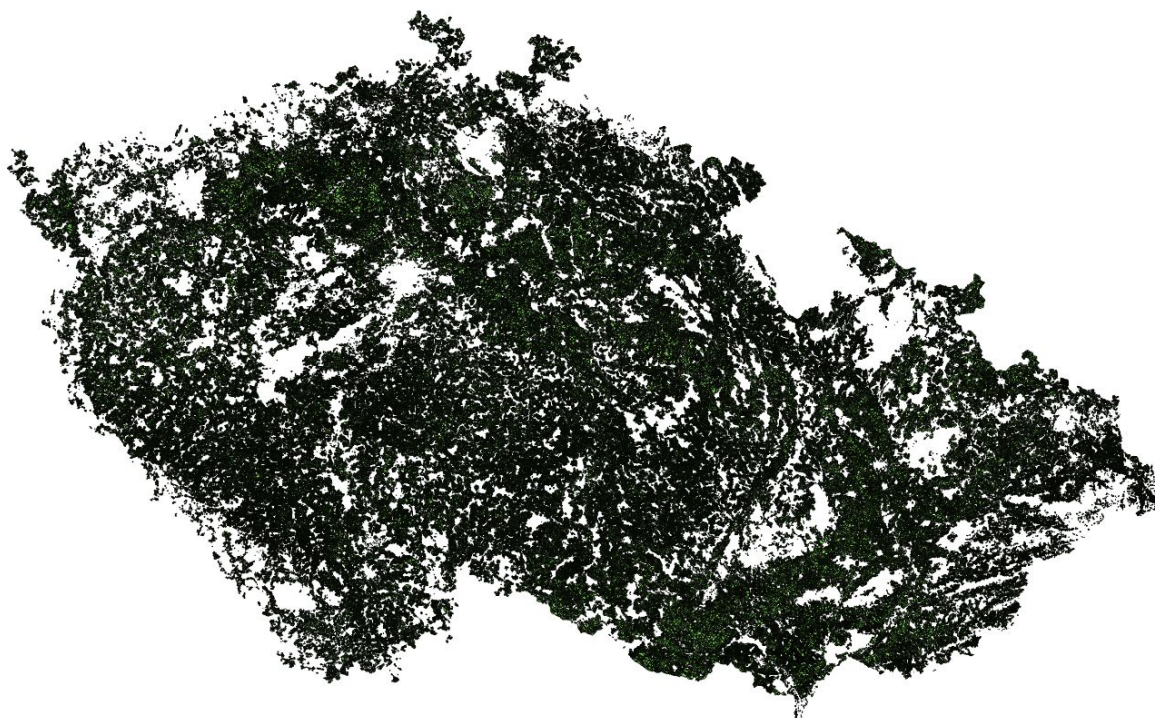
Pro usnadnění čitelnosti vrstev je přiložena tabulka vysvětlující jednotlivé zkratky.

Zkratka	Význam
PB	půdní bloky
DPB	díly půdních bloků
EVP	ekologicky významný prvek
OPV	osázená plocha vinic
ZOD	zranitelná oblast dusitany

2.1.2 Spouštění programu

1. Příklad na stažení dat z vrstvy DPB_EROZE_WFS pro celou ČR s nastavením velikosti strany dlaždice na 4 kilometry. V adresáři se zdrojovým kódem se vytvoří soubor s příponou gml, který je pojmenován podle názvu stahované vrstvy.

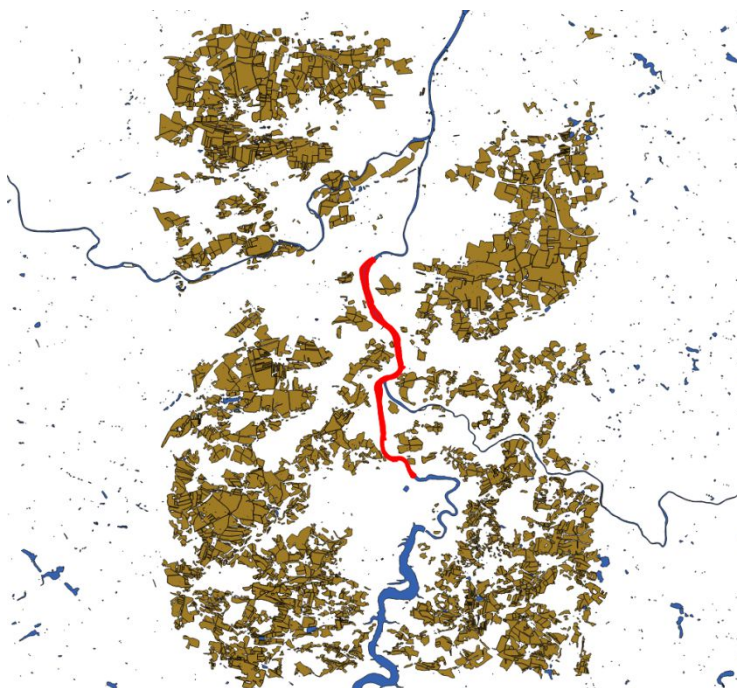
\$python3 LPIS.py DPB_EROZE_WFS 4000



Vrstva DPB_EROZE_WFS

2. Příklad na stažení dat z vrstvy DPB_EROZE_WFS pro území, které leží v okolí 10 kilometrů od vodní nádrže Vrané. Území definováno pomocí bboxu.

```
$ python3 LPIS.py DPB_EROZE_WFS 4000 --Bbox -758000 -1079000 -738000 -1048000
```



Vrstva DPB_EROZE_WFS definovaná bbox parametrem

Poznámka: Velikost dlaždice nutno volit s uvážením hustoty dat pro danou vrstvu. Při špatné volbě se nemusejí stáhnout kompletní data.

2.2 Tvorba dlaždic

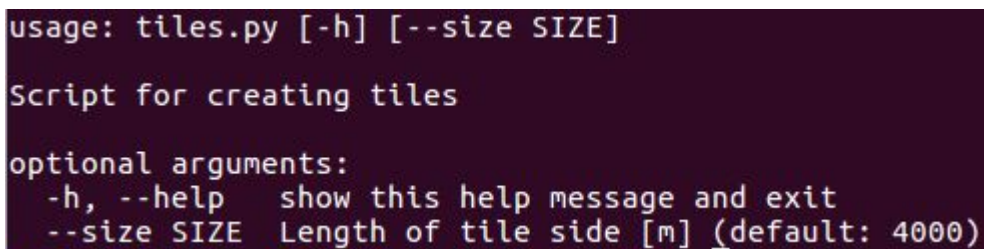
Program vytváří v adresáři se zdrojovým kódem soubor `tiles.gml`, který obsahuje geometrii jednotlivých dlaždic spolu s jejich id, které slouží jako identifikátor pro vyhledávání v programu `LPIS.py` v případě volby `--Tiles`. Pro správnou funkci programu `LPIS.py` je tedy nutné, aby velikost strany dlaždice byla v obou případech stejná.

2.2.1 Nápověda

Program není závislý na verzi jazyka python. Nápovědu k argumentům programu spustíme pomocí příkazu

```
$python3 tiles.py -h
```

Po zadání příkazu vrací program uživateli následující informace



```
usage: tiles.py [-h] [--size SIZE]

Script for creating tiles

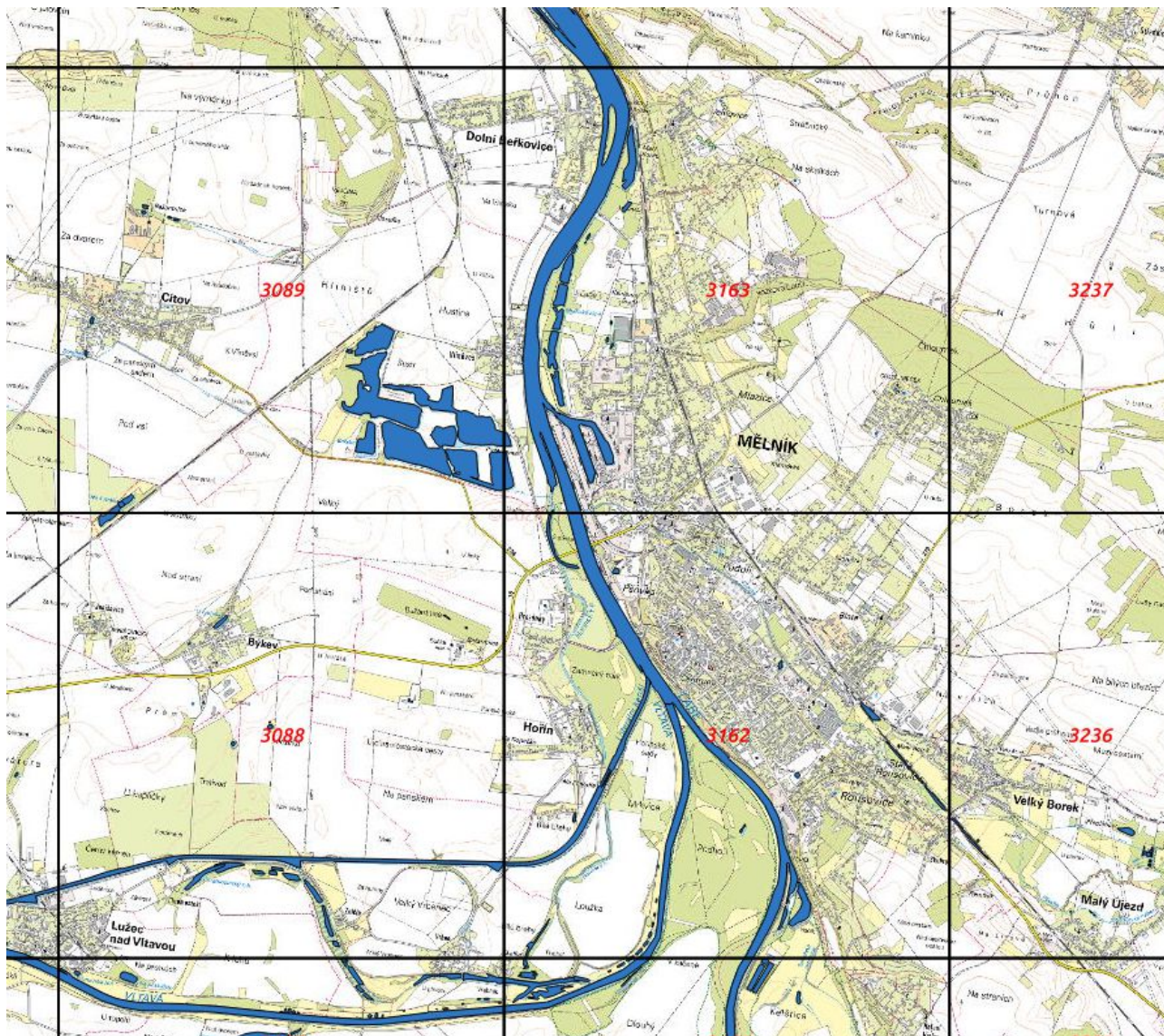
optional arguments:
  -h, --help  show this help message and exit
  --size SIZE Length of tile side [m] (default: 4000)
```

Program má pouze jeden volitelný parametr `-size`, který určuje velikost strany dlaždice v metrech a je přednastaven na hodnotu 4000.

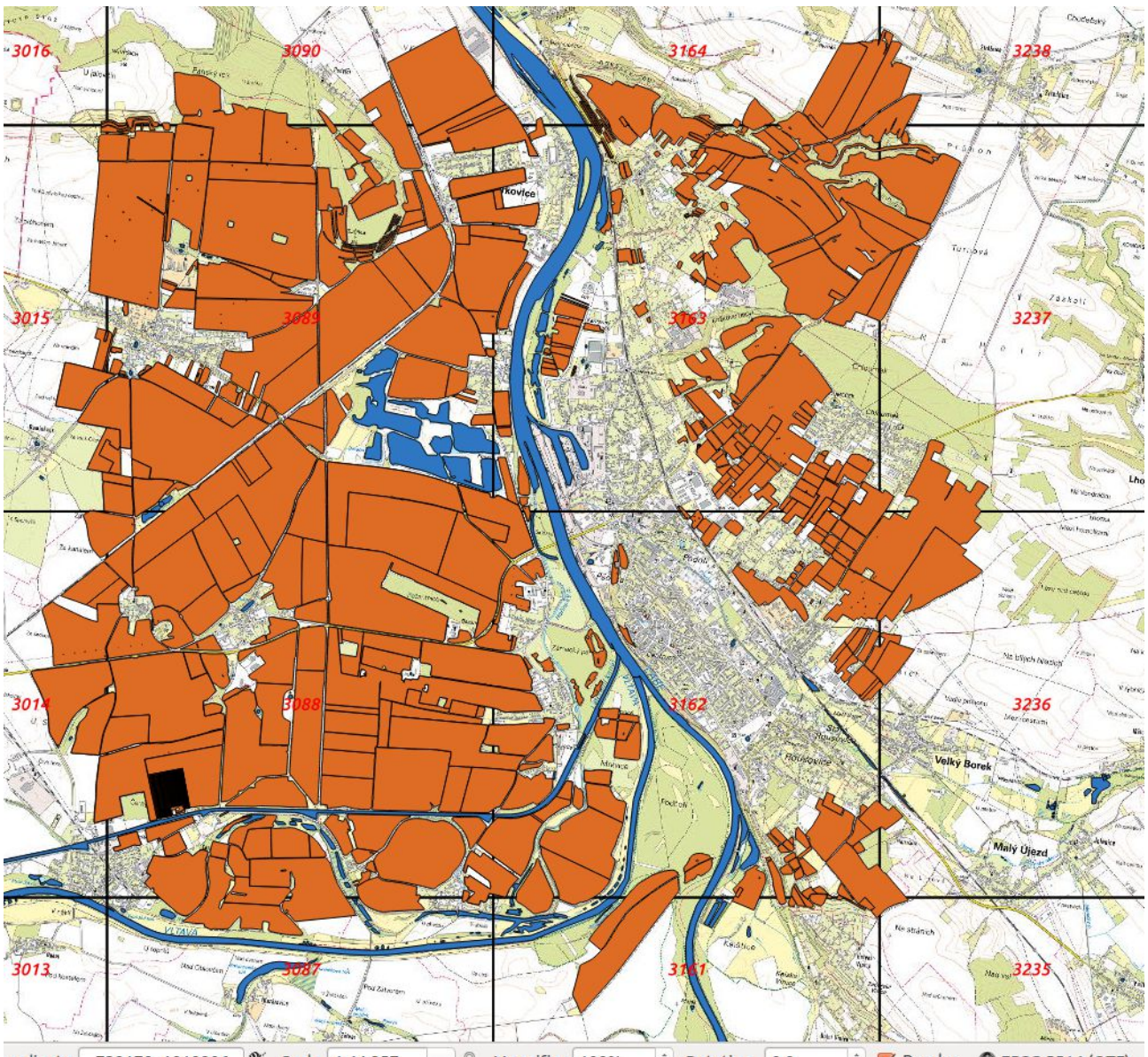
2.2.2 Spouštění programu

1. Příklad na tvorbu dlaždic o velikosti strany 4000. Po nahrání dlaždic do PostGIS a importu do QGIS (kapitola 3) se rozhodneme stáhnout data z vrstvy DPB_EROZE_WFS pro oblast v okolí Mělníka definovanou id dlaždic 3089 3163 3088 3162.

\$python3 tiles.py



\$python3 LPIS.py DPB_EROZE_WFS 4000 --Tiles 3089 3163 3088 3162



3 VRSTVY

3.1 Výběr vrstev

Polygonové

```
DPB_EROZE_WFS
ZOD_WFS
VYNOSOVE_HLADINY_BPEJ_WFS
```

3.2 Nahrání vrstev do postGIS

Stažená data ve formátu gml byly do PostGIS nahrány přes příkaz ogr2ogr z knihovny GDAL.

```
ogr2ogr -f "PostgreSQL" PG:"host="" port="" dbname="" user="" password="" nazev_souboru.gml  
-a_srs 'EPSG:5514' -nln nazev_tabulky;
```

3.3 Odstranění duplicitních prvků

Nahráná data bylo nutno zbavit duplicitních prvků. Pro odstranění duplicit z vrstvy byl použit následující příkaz.

```
DELETE FROM DPB_EROZE_WFS  
WHERE ogc_fid IN (SELECT ogc_fid  
FROM (SELECT ogc_fid,  
ROW_NUMBER() OVER (partition BY gml_fid ORDER BY ogc_fid) AS rnum  
FROM DPB_EROZE_WFS) t  
WHERE t.rnum > 1);
```

3.4 Vytvoření nové vrstvy

Po odstranění duplicit byly vytvořeny nové vrstvy.

```
create table erozni ohrozenost as  
select id::integer, wkb_geometry as geom, ctverec, zkod as zkraceny_kod, id_uz::integer as  
id_uzivatel, kultura_kod,  
seo_sum::float, seomeo_sum::float, neo_sum::float, erozohro as erozni_ohrozenost,  
sklonitost::float from DPB_EROZE_WFS;
```

```
create table zranitelne oblastidusitany as select ogc_fid as id, wkb_geometry as geom, idnitobl as  
id_oblasti from ZOD_WFS;
```

```
create table vynosove hladiny as select id_primarykey as id, wkb_geometry as geom, vynhled as  
vynosove_hladiny from VYNOSOVE_HLADINY_BPEJ_WFS
```

3.5 Vytvoření primárního klíče a prostorového indexu

Nad novými vrstvami byl vytvořen primární klíč a prostorový index. Tvorba probíhala nad všemi vrstvami stejným způsobem. První příkaz vytváří primární klíč nad zvolenou vrstvou druhý potom prostorový index.

```
alter table nazev_vrstvy add primary key(id);

create index on nazev_vrstvy using gist(geom);
```

4 PROSTOROVÉ DOTAZY

1. Seřadte okresy ČR sestupně podle velikosti zemědělské půdy. Výsledek zaokrouhlete na celé hektary. Uvažujte plochy, které celou svoji plochou náleží příslušnému okresu

```
select o.nazev as okres,
floor(sum(st_area(c.geom))*1e-4) as suma
from ruian.okresy as o
join uzpd17_e.eroznihozenost as c
on st_contains(o.generalizovanehranice,c.geom)
group by o.nazev order by suma desc;
```

okres	suma
Znojmo	94409.0
Třebíč	82611.0
Jindřichův Hradec	75843.0
Olomouc	74127.0
Klatovy	74000.0
České Budějovice	73835.0
Žďár nad Sázavou	73174.0
...	

[70 rows]

2. Vypište prvních deset okresů, kde je největší výměra silně erozně ohrožených pozemků. Vypište jméno okresu, velikost zemědělské půdy, velikost zemědělské půdy se silnou erozí ohrožeností a procentuální poměr ohrožené půdy vůči celku. Velikosti uvádějte v celých hektarech.

```
create temporary table seo_okresy as
select o.nazev as okres, floor(sum(e.seo_sum)) as seo
from erozniohrozenost as e
join ruian.okresy as o
on st_contains(o.generalizovanehranice, e.geom)
where e.erozni_ohrozenost = 'SEO'
group by o.nazev;
```

```
create temporary table plocha_okresy as
select o.nazev as okres,
floor(sum(st_area(c.geom))*1e-4) as suma
from ruian.okresy as o
join uzpd17_e.erozniohrozenost as c
on st_contains(o.generalizovanehranice, c.geom)
group by o.nazev;
```

```
select o.okres, o.suma as vymera_celek, s.seo,
floor((s.seo/o.suma)*100) as procenta from plocha_okresy
as o join seo_okresy as s on o.okres = s.okres
order by s.seo desc
limit 10;
```

okres	vymera_celek	seo	procenta
Bruntál	58608.0	6648.0	11.0
Šumperk	45659.0	6245.0	13.0
Vsetín	25243.0	5596.0	22.0
Klatovy	74000.0	5217.0	7.0
Zlín	32492.0	4607.0	14.0
Uherské Hradiště	44331.0	4515.0	10.0
Hodonín	52028.0	3382.0	6.0
Ústí nad Orlicí	61088.0	3306.0	5.0
Nový Jičín	44594.0	2540.0	5.0
Opava	57239.0	2252.0	3.0

3. Pro každý kraj vyberte největšího vlastníka pozemků. Vypište název kraje, id uživatele, rozlohu vlastněných pozemků zaokrouhlenou na celé hektary.

```
create temporary table kraj_uzivatele as
select k.nazev as kraj, id_uzivatel, count(e.id_uzivatel) as pozemky,
floor(sum(st_area(e.geom))*1e-4) as suma
from erozniohrozenost as e
join ruian.kraje as k
on st_contains(k.generalizovanehranice, e.geom)
group by k.nazev, id_uzivatel;

select k.kraj, k.id_uzivatel, k.pozemky as pozemky_sum, k.suma as rozloha
from (
select kraj, max(pozemky) as maxp
from kraj_uzivatele group by kraj
) as x
inner join kraj_uzivatele as k
on k.kraj= x.kraj
and k.pozemky=x.maxp;
```

kraj	id_uzivatel	pozemky_sum	rozloha
Jihočeský	29224	519	3702.0
Jihomoravský	32319	1223	9194.0
Praha	20999	82	522.0
Severočeský	43326	321	1460.0
Severomoravský	19662	661	2843.0
Středočeský	34020	692	4457.0
Východočeský	7	571	2535.0
Západočeský	42906	560	4472.0

5 ZÁVĚR

Cílem dokumentace je popsat způsob práce se dvěma vytvořenými programy v programovacím jazyce Python. Práce dále popisuje tvorbu několika tematických vrstev stažených pomocí vytvořeného programu z LPIS přes WFS. V době zpracování projektu nebylo možné pro většinu datových vrstev nacházejících se na LPIS stáhnout pomocí skriptu kompletní data pro celou ČR. Je tak možné, že některé vrstvy nejsou přes WFS poskytovány v celistvé podobě. Nad vytvořenými vrstvami bylo vytvořeno několik prostorových atributových dotazů.