

# Print list of tags to excel for manual inspection

This workflow reads in downloaded POI data and prints out information about downloaded POIs into an Excel file. Such excel was used for manual inspection of the data (by non-gis experts).

- [https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to\\_excel.html](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_excel.html)
- Note using `to_excel` requires <https://anaconda.org/conda-forge/xlsxwriter>

```
In [1]: import os
import glob
import pandas as pd
import geopandas as gpd
import matplotlib.pyplot as plt
```

```
In [2]: out_folder = r"..\results\downloaded_pois"
```

```
In [3]: # List geopackages that contain the downloaded OSM data
files = glob.glob(os.path.join(out_folder, "*gpkg"))
```

```
In [4]: categories = ["restaurants", "cafes", "pubs_clubs", "kindergartens",
                    "hotels", "museums", "theatres", "libraries", "cinemas",
                    "pharmacies", "kiosks",
                    "grocery_stores", "department_stores"]

summary = pd.DataFrame(index=["Joensuu", "Kuopio", "Lappeenranta", "Tampere"], columns=c
```

```
In [5]: summary
```

```
Out[5]:
```

	restaurants	cafes	pubs_clubs	kindergartens	hotels	museums	theatres	libraries	cinemas
Joensuu	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Kuopio	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Lappeenranta	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Tampere	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

```
In [6]: for gpkg in files:

    city = os.path.basename(gpkg).split("_")[1][:-5]

    # Define layers to include
    for layer in categories:

        data = gpd.read_file(gpkg, layer=layer)

        #update count to summary table
        summary.loc[city, layer] = len(data)
```

```

data = data[['name', 'name_fi', 'amenity', 'shop', 'tourism', 'opening_hours',
'addr_city', 'addr_country', 'addr_postcode', 'addr_street', 'website',
'source']]

#Add columns for manual inspection.
data["OK"] = ""
data["VANHA"] = ""
data["UUSI"] = ""
data["DUPLIKAATTI"] = ""
data["KORJATTU_OSMIIN"] = ""
data["HUOM"] = ""
data["RIVINUMERO"] = data.index

out_fp = r"..\\results\\downloaded_pois"
fp=os.path.join(out_fp, f"OSM_{city}.xlsx")

#Initiate excel with restaurants sheet:
if layer == "restaurants":
    with pd.ExcelWriter(fp) as writer:
        data.to_excel(writer, sheet_name=layer)

# Write other tags to new sheets
else:
    with pd.ExcelWriter(fp, mode="a") as writer:
        data.to_excel(writer, sheet_name=layer)

print("Done:", fp)

```

D:\ProgramFiles\Anaconda3\envs\python-gis\lib\site-packages\geopandas\geodataframe.py:422: RuntimeWarning: Sequential read of iterator was interrupted. Resetting iterator. This can negatively impact the performance.

```

for feature in features_lst:
Done: ..\\results\\downloaded_pois\\OSM_Joensuu.xlsx
Done: ..\\results\\downloaded_pois\\OSM_Kuopio.xlsx
Done: ..\\results\\downloaded_pois\\OSM_Lappeenranta.xlsx
Done: ..\\results\\downloaded_pois\\OSM_Tampere.xlsx

```

```

In [7]: import datetime
today = datetime.date.today().strftime("%Y_%m_%d")

```

```

In [8]: summary.T.to_csv(os.path.join(out_folder, f"downloaded_pois_summary_{today}.csv"))

```

```

In [9]: summary.T

```

```

Out[9]:

```

	Joensuu	Kuopio	Lappeenranta	Tampere
<b>restaurants</b>	118	157	103	446
<b>cafes</b>	53	37	34	136
<b>pubs_clubs</b>	30	32	23	152
<b>kindergartens</b>	60	52	33	119
<b>hotels</b>	15	22	12	30
<b>museums</b>	8	20	14	36

	Joensuu	Kuopio	Lappeenranta	Tampere
<b>theatres</b>	7	7	2	16
<b>libraries</b>	14	17	13	41
<b>cinemas</b>	1	3	1	3
<b>pharmacies</b>	13	17	12	41
<b>kiosks</b>	9	21	10	46
<b>grocery_stores</b>	56	63	41	202
<b>department_stores</b>	12	8	4	26