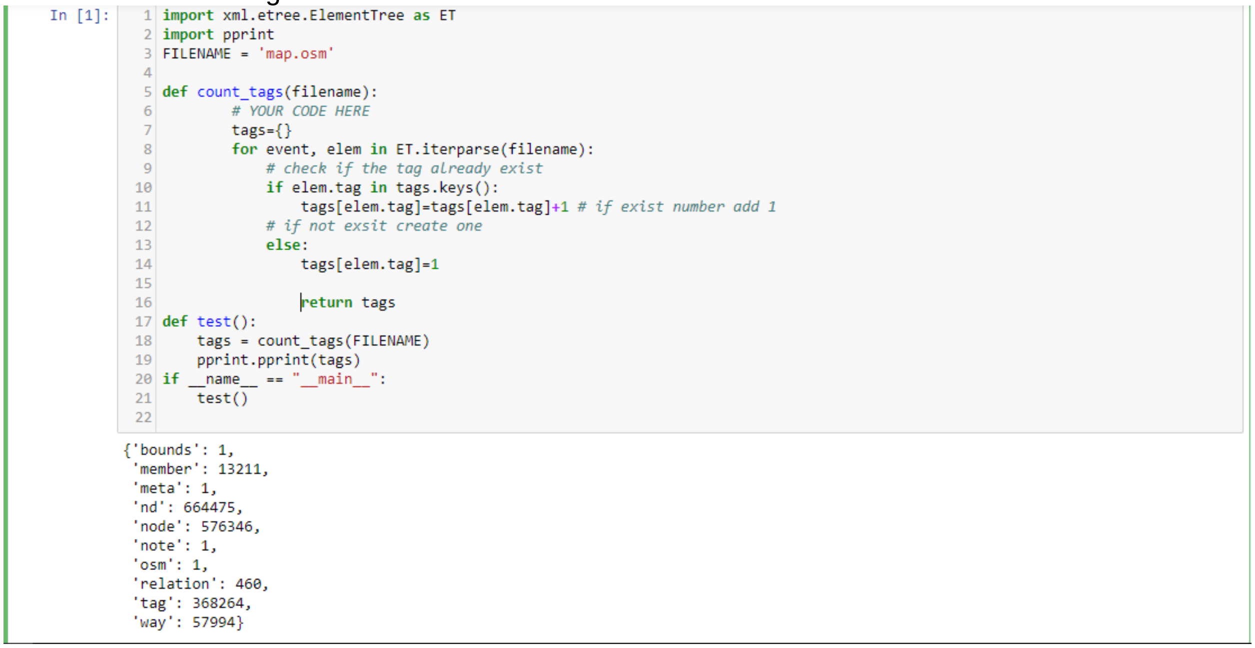
OpenStreetMap Sample Project

Data Wrangling with MongoDB

Lucas Mwai

Map Area: Marietta, GA, United States
https://www.openstreetmap.org/export#map=12/33.9486/-84.5425
Overpass API node (-84.7169,34.0316, -84.3681,33.8656)

An overview of the tags used in the dataset



```
import xml.etree.ElementTree as ET
import pprint
FILENAME = 'map.osm'

def count_tags(filename):
        # YOUR CODE HERE
        tags={}
        for event, elem in ET.iterparse(filename):
            # check if the tag already exist
            if elem.tag in tags.keys():
                tags[elem.tag]=tags[elem.tag]+1 # if exist number add 1
            # if not exsit create one
            else:
                tags[elem.tag]=1

                return tags
def test():
    tags = count_tags(FILENAME)
    pprint.pprint(tags)
if __name__ == "__main__":
    test()

```

```
{'bounds': 1,
 'member': 13211,
 'meta': 1,
 'nd': 664475,
 'node': 576346,
 'note': 1,
 'osm': 1,
 'relation': 460,
 'tag': 368264,
 'way': 57994}
```

1. Problems Encountered in the Map

After downloading the map data and did some data wrangling in python, I noticed the following problems

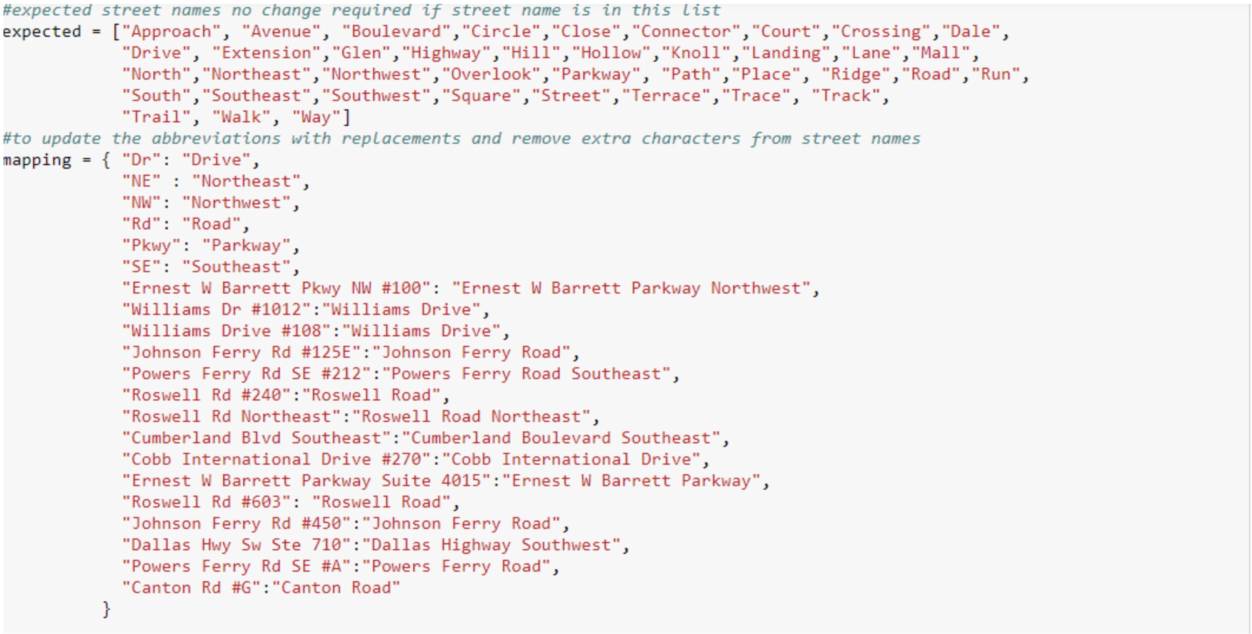
I)mixing abbreviations and full words in the street names

II)addition of extra characters in the street names

Mixing abbreviations and full words in the street names

A combination of Full names in some cases while in others abbreviations were used. examples include (Road/Rd, Drive/Dr, Parkway/Pkwy) among others. I replaced the abbreviations with full names to ensure consistency in the naming convection

A few examples to illustrate the change
```
Earnst W. Barrett Pkwy changed to  Earnst W. Barrett Parkway
Roswell Rd changed to Roswell Road
```



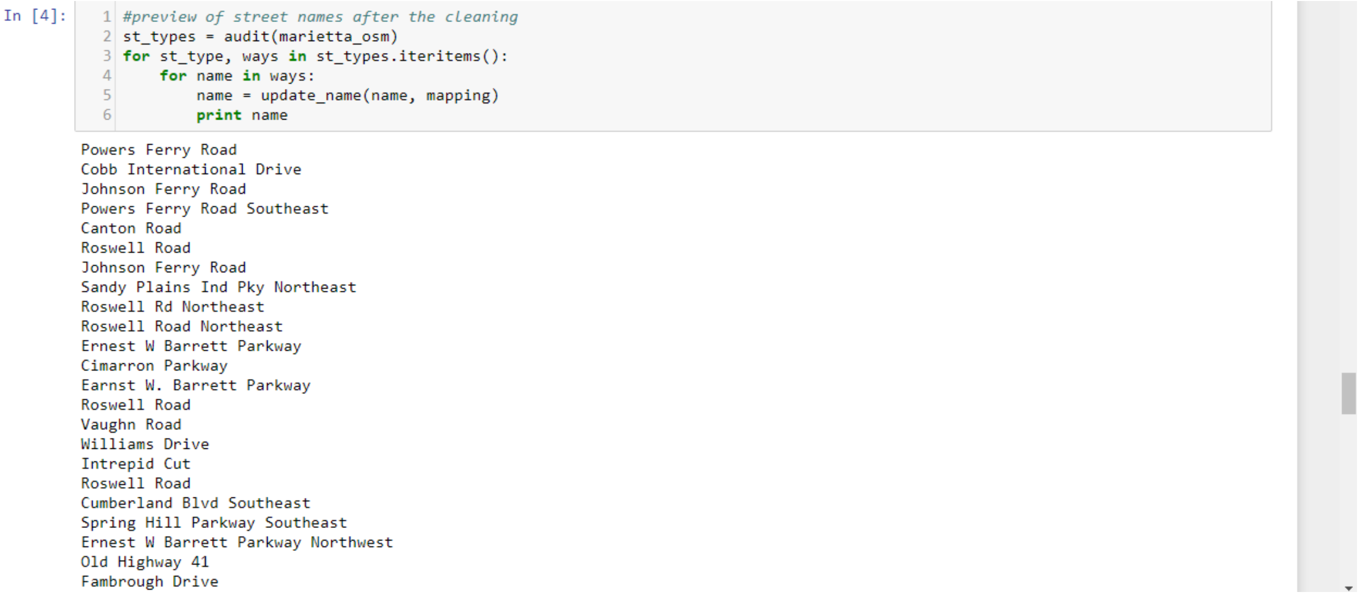
```
#expected street names no change required if street name is in this list
expected = ["Approach", "Avenue", "Boulevard","Circle","Close","Connector","Court","Crossing","Dale",
            "Drive", "Extension","Glen","Highway","Hill","Hollow","Knoll","Landing","Lane","Mall",
            "North","Northeast","Northwest","Overlook","Parkway", "Path","Place", "Ridge","Road","Run",
            "South","Southeast","Southwest","Square","Street","Terrace","Trace", "Track",
            "Trail", "Walk", "Way"]
#to update the abbreviations with replacements and remove extra characters from street names
mapping = { "Dr": "Drive",
            "NE" : "Northeast",
            "NW": "Northwest",
            "Rd": "Road",
            "Pkwy": "Parkway",
            "SE": "Southeast",
            "Ernest W Barrett Pkwy NW #100": "Ernest W Barrett Parkway Northwest",
            "Williams Dr #1012":"Williams Drive",
            "Williams Drive #108":"Williams Drive",
            "Johnson Ferry Rd #125E":"Johnson Ferry Road",
            "Powers Ferry Rd SE #212":"Powers Ferry Road Southeast",
            "Roswell Rd #240":"Roswell Road",
            "Roswell Rd Northeast":"Roswell Road Northeast",
            "Cumberland Blvd Southeast":"Cumberland Boulevard Southeast",
            "Cobb International Drive #270":"Cobb International Drive",
            "Ernest W Barrett Parkway Suite 4015":"Ernest W Barrett Parkway",
            "Roswell Rd #603": "Roswell Road",
            "Johnson Ferry Rd #450":"Johnson Ferry Road",
            "Dallas Hwy Sw Ste 710":"Dallas Highway Southwest",
            "Powers Ferry Rd SE #A":"Powers Ferry Road",
            "Canton Rd #G":"Canton Road"
          }
```

Addition of extra characters in the street names

Addition of letters to street names, examples include addition of #G and #A within the street name. There is also some addition of numbers at the end of street names.
Although some roads have numbers in their name, a closer look seemed to show they are just repetitions caused by users who input the whole physical address to the street names. I removed the additional characters and the extra numbers. Here is a few examples to illustrate the change

(['Canton Rd #G']) changed to Canton Road

(['Ernest W Barrett Pkwy NW #100']), changed to Ernest W Barrett Pkwy Northwest
 (['Williams Dr #1012']), changed to Williams Drive



In [4]:

```
#preview of street names after the cleaning
st_types = audit(marietta_osm)
for st_type, ways in st_types.iteritems():
    for name in ways:
        name = update_name(name, mapping)
        print name
```

```
Powers Ferry Road
Cobb International Drive
Johnson Ferry Road
Powers Ferry Road Southeast
Canton Road
Roswell Road
Johnson Ferry Road
Sandy Plains Ind Pky Northeast
Roswell Rd Northeast
Roswell Road Northeast
Ernest W Barrett Parkway
Cimarron Parkway
Earnst W. Barrett Parkway
Roswell Road
Vaughn Road
Williams Drive
Intrepid Cut
Roswell Road
Cumberland Blvd Southeast
Spring Hill Parkway Southeast
Ernest W Barrett Parkway Northwest
Old Highway 41
Fambrough Drive
```

2. Data Overview

File sizes

Marietta map osm .... 129 MB

Marietta map osm json .... 188 MB

Number of documents

```
> db.mapdata.find().count()
```

```
   634344
```

Number of nodes

```
> db.mapdata.find({"type":"node"}).count()
```

```
576348
```

Number of ways

```
db.mapdata.find({"type":"way"}).count()
```

```
57992
```

Number of unique users

```
db.mapdata.distinct("created.user").length
```

```
431
```

Top 1 contributing user

```
> db.mapdata.aggregate([     {"$match":{"type":"node"}},
    {"$group":{"_id":"$created.user","count":{"$sum":1}}},
   {"$sort":{"count":-1}},     {"$limit":1} ])
```

```
{ "_id" : "Saikrishna_FultonCountyImport", "count" : 172555 }
```

Top amenities

```
> db.mapdata.aggregate([
...     {$match:{"amenity":{"$exists":1},"type":"node"}},
...     {"$group":{"_id":"$amenity","count":{"$sum":1}}},
...     {$sort:{"count":-1}},
...     {"$limit":10}
... ])
{ "_id" : "restaurant", "count" : 131 }
{ "_id" : "atm", "count" : 119 }
{ "_id" : "place_of_worship", "count" : 89 }
{ "_id" : "grave_yard", "count" : 85 }
{ "_id" : "school", "count" : 63 }
{ "_id" : "fast_food", "count" : 38 }
{ "_id" : "bench", "count" : 24 }
{ "_id" : "post_box", "count" : 23 }
{ "_id" : "cafe", "count" : 23 }
{ "_id" : "parking_entrance", "count" : 20 }
```

3. Additional Considerations

It seems that users that different ways of referring to amenities in the food category, I notice that users use categories restaurant, fast-food and cafe to refer to different places to eat, this information seems to be overlapping

A more direct query that narrows the category might provide better results for example finding unique names in all three categories of the food amenity.

```
> db.mapdata.aggregate([{"$match":{"amenity": {"$in": ["restaurant",
               "cafe", "fast_food", ]}}},
               {"$match": {"name": {"$exists": 1}}},
               {"$group": { "_id": "$name","amenity": {"$first": "$amenity"},
               "cuisine": {"$push": "$cuisine"},"count": {"$sum": 1}
               }},{"$project": { "_id": 0,"count": 1,"name": "$_id", "type" : {"$concat":
               ["$amenity", " - ", {"$ifNull": [{"$arrayElemAt": ["$cuisine", 0 ]}, "unknown cuisine"] }
               ]}}},{"$sort": {"count": -1}}, {"$limit": 20}     ])
```

```
{ "count" : 21, "name" : "Subway", "type" : "fast_food - sandwich" }
{ "count" : 20, "name" : "McDonald's", "type" : "fast_food - burger" }
{ "count" : 18, "name" : "Waffle House", "type" : "restaurant - diner" }
{ "count" : 11, "name" : "Wendy's", "type" : "fast_food - burger" }
{ "count" : 11, "name" : "Starbucks", "type" : "cafe - coffee_shop" }
{ "count" : 9, "name" : "Chick-fil-A", "type" : "fast_food - chicken" }
{ "count" : 9, "name" : "Burger King", "type" : "fast_food - burger" }
{ "count" : 8, "name" : "Zaxby's", "type" : "fast_food - chicken" }
{ "count" : 7, "name" : "Taco Bell", "type" : "fast_food - mexican" }
{ "count" : 6, "name" : "IHOP", "type" : "restaurant - pancakes" }
{ "count" : 5, "name" : "Arby's", "type" : "fast_food - sandwich" }
{ "count" : 4, "name" : "Dunkin' Donuts", "type" : "cafe - donuts" }
{ "count" : 4, "name" : "Ted's Montana Grill", "type" : "restaurant - american" }
{ "count" : 3, "name" : "Firehouse Subs", "type" : "restaurant - sandwich" }
{ "count" : 3, "name" : "Mellow Mushroom", "type" : "restaurant - pizza" }
{ "count" : 3, "name" : "Little Caesars", "type" : "fast_food - pizza" }
{ "count" : 2, "name" : "Dairy Queen", "type" : "fast_food - burger" }
{ "count" : 2, "name" : "J. Christopher's", "type" : "restaurant - american" }
{ "count" : 2, "name" : "Krystal", "type" : "fast_food - burger" }
{ "count" : 2, "name" : "Mezza Luna Pasta & Seafood", "type" : "restaurant - italian" }
```

Other ideas about the dataset

There seems to be some inconsistencies in various categories within the tags in the dataset, I only explored the food category but I would imagine that there are more of them in the dataset

Since the users are using different ways to refer to the amenities in the food category, it would be ideal to give guidance to users on how to enter the data in this category. Making available a specific outline and rules for users to organize this category instead of it being broadly in the food amenity. Utilizing automatic ways like scripts to validate the data entered would also bring some consistency. Constant checking, cleaning and validation would ensure that the data is always accurate

The benefits of this validation would mean that the data is cleaner and more accurate and would not require the long processes for cleaning it.

There are challenges arising from these potential solutions, the fact that this is an open source forum may make it quite complicated to enforce the following of the outline provided. Users would have to have the discipline to ensure they follow the guidance for inputting data but there is no way to enforce these rules, using automatic scripts for validation would mean there would be enormous workload done in the backend to ensure this validation is effective.

Religion

This area in the south of the united states known as the bible belt, Christian churches are almost 98% of the total religious places of worship

```
>  db.mapdata.aggregate([     {"$match":{"amenity":"place_of_worship","type":"node"}},
{"$group":{"_id":"$religion","count":{"$sum":1}}},     {"$sort":{"count":-1}},     {"$limit":5} ])
```

```
{ "_id" : "christian", "count" : 87 }

{ "_id" : "jewish", "count" : 2 }
```

Conclusion

In this exercise, I did some data wrangling and made some corrections to the map file., it was a lot of challenging work but I was excited to work on it and especially being that I'm currently residing here, it made the corrections more relatable since I have driven on some of these streets and roads. These corrections are however not necessarily complete, it does require more wrangling and corrections to improve the maps and this exercise was just a portion of that. Although no data can be claimed as perfect, continued improvements should make changes more accurate and reliable