OpenStreetMap Data

Map Area

- San Francisco, CA, United States
- https://mapzen.com/data/metro-extracts/metro/san-francisco_california/ (https://mapzen.com/data/metro-extracts/metro/san-francisco_california/)

I chose the map of San Francisco for this project because I have never been in San Francisco and I don´t even live in United States. So, my purpose is to evaluate how much of information I can learn from this data, how many inconsistencies I can find without knowing the place.

Problems Encountered in the Map

Although the map area is of San Francisco, the file comprehend a larger area than the city of San Francisco, including some information of San Mateo, Berkeley, Oakland and so on. The main problems I found in the map are:

- **Problematic street names:** Instead of Avenue, Street, Road, the street name is a number or an anormal character, like #105, #155, 122°29'07.1;
- **Problematic city names:** Instead of San Francisco, or other city, the city name is a number, like 11720, 155, 157;
- **Zip Code:** probably wrong zip codes, like 515, 1087, 2952, since the most common zip code contains 5 digits and it starts with 94;
- **State name:** Some tags have the state name as a number, like 1463-1465, instead of CA;
- **"node" tags:** Some "node" tags do not have user and uid field;

The problems I was sure that were a typo, I corrected before saving the information in the csv file, inside the shape_element function. So, I wrote some fuctions (update_zip_code, update_name, update_city, update_state and update_country) in order to correct the fields. The "nodes" tags which doesn´t have user, uid or other field, I did not save in the csv file. The problematic zip codes with less than 5 digits were excluded. However, if the zip code had more than 5 digits, like 94045-0809, only the first 5 digits were kept.

Database and Tables

Once I saved the information into a csv file, I created the Database and five Tables from them.

Data Overview and Additional Ideas

File Sizes

```
San-Francisco_California.osm .......961 MB
project3.sqlite ....................528 MB
nodes.csv ..........................380 MB
nodes_tags.csv .......................9 MB
ways.csv ............................31 MB
ways_tags.csv .......................50 MB
ways_nodes.csv .....................133 MB
```

Number of Nodes and Ways

Query = "SELECT COUNT(*) FROM nodes;"
Query_1 = "SELECT COUNT(*) FROM ways;"

- Number of nodes: 4581032
- Number of ways: 532313

Number of Unique Users

Query = "SELECT COUNT(DISTINCT(all_nodes.uid)) FROM (SELECT uid FROM nodes UNION ALL SELECT uid FROM ways) as all_nodes;"

- Number of unique users: 2566

10 Users that Most Contributed

Query = "SELECT all_nodes.user, COUNT(*) as count FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) as all_nodes GROUP BY all_nodes.user ORDER BY count DESC limit 10;"

```
ediyes       918915
Luis36995    710132
Rub21        395225
RichRico     224394
calfarome    185130
oldtopos     167544
KindredCoda  151716
karitotp     134937
samely       125525
abel801      108315
```

10 Most Common Cities

Query = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT * FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key=='city' GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
Redwood City   23527
San Francisco  17208
Berkeley        5626
Piedmont        3812
Palo Alto       1642
Oakland         1378
Richmond        1354
Union City       263
Albany           223
Burlingame       199
```

10 Most Common Zipcodes

QUERY = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT* FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key='postcode'GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
94122 5106
94611 2990
94116 2202
94610 1357
94117 1219
94133 1096
94103 797
94127 705
94109 452
94063 383
```

10 Most Common Streets

QUERY = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT* FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key='street'GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
Irving Street    731
Page Street      548
9th Avenue       544
Broadway         462
10th Avenue      455
14th Avenue      432
El Camino Real   431
12th Avenue      394
8th Avenue       390
Funston Avenue   383
```

Top 10 types of Amenities

QUERY = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT* FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key='amenity'GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
parking           4506
restaurant        3124
school            1312
bench             1155
place_of_worship  1154
cafe               988
fast_food          682
post_box           677
bicycle_parking    560
toilets            492
```

Popular Cuisines

QUERY = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT* FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key='cuisine'GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
mexican       279
coffee_shop   261
pizza         213
burger        195
chinese       189
japanese      150
italian       143
sandwich      141
american      133
thai          113
```

Important Sources of Information

QUERY = "SELECT all_tags.value, COUNT(*) as count FROM (SELECT* FROM nodes_tags UNION ALL SELECT * FROM ways_tags) as all_tags WHERE all_tags.key='source'GROUP BY all_tags.value ORDER BY count DESC limit 10;"

```
City of Redwood City, CA 1013             23305
tiger_import_dch_v0.6_20070809            17050
Bing                                      11126
bing                                       7504
EarthScope (http://www.earthscope.org), International Solar Information Solutions (
http://www.isi-solutions.org), OpenTopography (http://www.opentopography.org) 4526
City of Palo Alto CA 0713                  2310
data.sfgov.org                             1870
photograph                                 1693
survey                                     1475
NHD                                        1430
```

Other Ideas about datasets

Since the data is updated or even constructed by contributors, it is difficult to insure the data quality. Some typos or errors are easier to fix, but it is not the case for all the fields. So, a guidance about how to update information could improve the quality of the information. For example, if the field is a city name, you should not put a number on it. However, some difficulties could occur when implementing this, for example:

- who is responsible for defining what is a valid pattern;
- If this valid pattern is appropriate for all places;
- how we assure this pattern is been followed;
- If the data comes from a GPS, different brands do not have the same format of data;

Conclusion

My challenge in this exercise was to discover how much of information I could learn from the data without knowing the place. My discoveries are:

- In the auditing process I found problematic street, cities, state and country names, as well as zip codes and phone numbers. Some of these were typos and I could corrected them, but others were not. I also found some tags who did not have user and uid fields.
- The map area include San Francisco and other cities like Palo Alto, San Carlos, Berkeley, and so on.
- There are many different users, 2566, and the user 'ediyes' was the biggest contributor. The two most common amenities are parkings and restaurants, and the most popular cuisine is Mexican.
- The three most important source of information are City of Redwood, Tiger GPS and Bing.

```
In [ ]:
```