# OpenStreetMap Data Analysis

This project involves analyzing OpenStreetMap data for a location and performing required data cleaning tasks for insertion into the SQL database.

## Location:

Seattle, WA, United States

https://www.openstreetmap.org/relation/237385

I chose this location as this is the city I currently live in, also as Seattle is in a state bordering another country, I was curious to look at the data for this region and would like to contribute my work to the OpenStreetMap Project.

## Problems Encountered:

While going through a sample of the XML data obtained through openstreetmap.org for Seattle, WA various problems and errors were identified. These have been listed below:

- Postal Codes: Some of the postal codes listed (V9B4R3, V8P2P4, etc) did not belong to the region being studied and some of the postal codes had area codes listed along with them (98124-1904, 98052-4176).
- Street Names: Various instances of street name abbreviation were identified (using E instead of East, SE instead of Southeast).
- County Names: It was observed that some county names contained in 'v' attributes under the sub-element of the 'way' with the 'k' attribute of 'tiger:county' had colons or semicolons within them, which separated two county names.
- Data Source: No standard/format is followed while mentioning the sources. This has led to entries such as 'Bing', 'Bing Imagery', 'Bing Imagery retrieved 9 Jul 2011' and so on. Also, some values contain colons, commas or semicolons within them which separate multiple sources.

### Dealing with problems in Data Source:

First identification of all possible source names in the OSM file were performed to determine the changes needed to be performed. It was seen that various values will have to be standardized and if there existed a semicolon, comma or an and (&), multiple sources were considered in addition of that element.

In order to clean the data, first a list of standard names is established to replace the current values. Various keywords (such as 'bing', 'yahoo', 'mapquest') and characters (',', ';', '&') are searched for in the existing source names to make this conversion and make sure no data is lost. This is performed using functions named check_source and conv_standard. A part of the check_source function is added below showing the process of conversion of value with a '&' to the standard values:

```
final = []

if(and_search.search(elem.attrib['v'])):
      vals = elem.attrib['v'].split('&')
      for i in range (len(vals)):
            vals[i] = vals[i].strip()
            vals[i] = conv_standard(vals[i])
            if(vals[i] in accepted_values):
                final.append(vals[i])
```

## Working with Postal Codes:

It is seen that postal codes like V9B4R3 and V8P2P4 are postal codes in British Columbia, Canada which is the neighboring state of Washington where Seattle is located. These postal codes will be removed from the data as they don't belong to the region being analyzed.

Also, certain postal codes have area codes within them separated by a hyphen. As the data needs to only be holding postal codes, the area codes are removed. This is performed by the function check_postalCodes:

```
def check_postalCodes(subelem):

    if((Alphabet_check.search(subelem.attrib['v']))):
        return 0
    elif(hyphen_check.search(subelem.attrib['v'])):
        hyph_sep = subelem.attrib['v'].split('-')
        return (hyph_sep[0])
    else:
        return (subelem.attrib['v'])
```

## Abbreviated Street Names:

The last part of the street names has been abbreviated in various entries. Thus, a mapping dictionary is written to map the abbreviated values with their actual values and changes are made based on this mapping by the check_street function.

### Checking County Names:

While analyzing county names, it was seen that these values are part of the sub-elements of the 'way' tag and the values separated by the colon or semicolon might describe a 'way' that connects multiple counties. In this case, the origin of the way i.e. the first value is kept and the other value is removed by the check_county function i.e. the following transformation takes place:

Before: Mason, WA:Thurston, WA

King, WA:Snohomish, WA

Thurston, WA;Lewis, WA

After: Mason, WA

King, WA

Thurston, WA

After cleaning has been performed, data is converted into .csv format and stored into five tables based on the schema in schema.py. These tables are then imported into SQL database for further analysis.

## Statistics of Computed Dataset:

### File Size:

| | |
|---|---|
| Seattle_washington.osm…… | 1677.976 MB |
| Data_seattle.db…… | 965.823 MB |
| Nodes.csv…… | 646.990 MB |
| Nodes_tags.csv…… | 42.838 MB |
| Ways.csv…… | 45.900 MB |
| Ways_nodes.csv…… | 201.022 MB |
| Ways_tags.csv…… | 122.018 MB |

### Number of Nodes:

```
SELECT COUNT(*) FROM nodes;
```

7775796

## Number of Ways:

```
SELECT COUNT(*) FROM ways;
```
772176

## Number of Unique Users:

- Contributed to Nodes:
  ```
  SELECT COUNT( DISTINCT(uid)) FROM nodes;
  ```
  3135

- Contributed to Ways:
  ```
  SELECT COUNT( DISTINCT(uid)) FROM ways;
  ```
  2396

## Top amenities in the region:

```
SELECT value, COUNT(value) as Numbers
FROM nodes_tags
GROUP BY value
HAVING key = 'amenity'
ORDER BY Numbers DESC
LIMIT 10;
```
value|Numbers
bicycle_parking|3348
bench|3088
restaurant|2791
waste_basket|1401
cafe|1365
fast_food|1196
school|876
parking|816
place_of_worship|734
toilets|733

It was surprising to see that Seattle has more bicycle parking spots than benches, waste baskets, restaurants, etc.

## Finding the number of Nodes with multiple Sources:

```
SELECT COUNT (*) FROM
(SELECT id, COUNT(*) as Sources FROM
(SELECT id, key, value FROM nodes_tags where key = 'source')
GROUP BY id
HAVING Sources>=2);
```
COUNT(*)
45744

Finding the users who have contributed the most to the amenity, 'restaurant':

```
SELECT nodes.uid, nodes.user, COUNT(*) as Contributions
FROM nodes JOIN nodes_tags ON nodes.id=nodes_tags.id
GROUP BY nodes.uid, nodes_tags.value
HAVING  nodes_tags.value = 'restaurant'
ORDER BY Contributions DESC
LIMIT 5;
```

uid|user|Contributions
1408522|Omnific|775
692614|seattlefyi|388
1529630|Brad Meteor|198
2601744|sctrojan79|197
2814663|Ballard OpenStreetMap|92

Some Important Statistics:

- 22.3 % of Nodes have multiple sources (if they have sources provided).
- There are approximately 10 times as many 'nodes' as there are 'ways'.
- Most common amenity found are bicycle parking spots which represent 14.3 % of all amenities listed.
- The top contributor to the dataset was the user with the name, 'Glassman' accounting for 14.9 % of the total nodes listed.

## Conclusions and Recommendations:

Based on this study, it can be concluded that the data for Seattle has been thoroughly cleaned and some interesting features about this region have been determined.

For future studies and analysis, the dataset could be expanded to mention the elevation of each node as this region has beaches as well as tall mountains. A particular format for addition of 'Sources' should also be developed to make sure entries follow the same pattern and redundant entries are not inputted into the database. This task can be challenging as multiple organizations will be having multiple tools, ways of reporting features which are being added onto the database. Also, a distinction could be added, which would tell if the data is obtained automatically from sources or added manually.