Project 3-> OpenStreetMap Data Wrangling with SQL

Name-> Prateek Pande

- **Map Area**-> I have chosen new data which is part of Central Delhi

1. Data Audit

Unique Tags

Looking at the XML file again I found that it uses different types of tags. So I parse the Central Delhi, India dataset using Element Tree and count number of the unique tags.
Uniquetags.py is used to count the numbers of unique tags.

- bounds-> 2
- member-> 2302
- nd-> 265044
- node-> 219586
- osm-> 1
- relation-> 542
- tag-> 60192
- way-> 40536

Patterns in the Tags

The "k" value of each tag contain different patterns. Using tagpattern.py I have counted each of four tag categories.

- "lower"-> 56168 for tags that contain only lowercase letters and are valid
- "lower colon"-> 3856 for otherwise valid tags with a colon in their names
- "problemchars"-> 0 for tags with problematic characters and
- "other"-> 168 for other tags that do not fall into the other three categories.

2. Problems Encountered in the Map

Street address inconsistencies

On checking the street names first I have seen that there are not much discrepancies in the date, few which i have corrected are mentioned below.

Ln -> Lane
lane-> Lane

City Name Inconsistencies

On checking the city name under node tag I found that there are lot of discrepancies in a manner that city name New Delhi was written in many different languages. I have corrected and converted all of them into English.

"Nieu-Delhi" ->"New Delhi",
"ኒው ዴሊ" ->"New Delhi",
"Nueva Delhi"->"New Delhi",
"Nīpe Delhi"->"New Delhi",
"دلهي الجديد"->"New Delhi",
"نيودلهي"->"New Delhi",
"دلهي جديد"->"New Delhi",
"Naujasės Delės"->"New Delhi",
"Нью-Дэлі"->"New Delhi",
"Нью-Дэлі"->"New Delhi",
"Ню Делхи"->"New Delhi",
"নতুন দিল্লি"->"New Delhi",
"ནིའུ་དེལ་ཧི།"->"New Delhi",

"নুৱা দিল্লী"->"New Delhi",
"Nova Delhi"->"New Delhi",
"Nueva Delhi"->"New Delhi",
"نیو دهلی"->"New Delhi",
"Nyu Deli"->"New Delhi",
"Nové Dillí"->"New Delhi",
"Delhi Newydd"->"New Delhi",
"Neu-Delhi"->"New Delhi",
"Delhiyo Newe"->"New Delhi",
"ނިއުދިއްލީ"->"New Delhi",
"Νέο Δελχί"->"New Delhi",
"Nov-Delhio"->"New Delhi",
"Nueva Delhi"->"New Delhi",
"Nueva Delhi"->"New Delhi",
"دهلی "و ن"->"New Delhi",
"Nij Delly"->"New Delhi",
"Nua-Deilí"->"New Delhi",
"Nova Deli - नई दिल्ली"->"New Delhi",
"Delhi Noa"->"New Delhi",
"ניו דלהי"->"New Delhi",
"नई दिल्ली"->"New Delhi",
"Niou Deli"->"New Delhi",
"Újdelhi"->"New Delhi",
"Նյու Դելի"->"New Delhi",
"Nove Delhi"->"New Delhi",
"Nova-Delhi"->"New Delhi",
"Nýja Delí"->"New Delhi",
"Nuova Delhi"->"New Delhi",
"ニューデリー"->"New Delhi",
"ნიუ-დელი"->"New Delhi",
"Нью-Дели"->"New Delhi",
"ಹೊಸ ದೆಹಲಿ"->"New Delhi",
"뉴델리"->"New Delhi",
"نیو دلهی"->"New Delhi",
"Delhiya Nû"->"New Delhi",
"Dellium Novum"->"New Delhi",
"Nei-Delhi"->"New Delhi",
"Neuva Delhi"->"New Delhi",
"Naujasis Delis"->"New Delhi",
"Ņūdeli"->"New Delhi",
"Њу Делхи"->"New Delhi",
"ന്യൂ ഡെൽഹി"->"New Delhi",
"Шинэ Дели"->"New Delhi",
"नवी दिल्ली"->"New Delhi",
"နယူးဒေလီမြို့"->"New Delhi",
"نو دهلی"->"New Delhi",
"Yancuīc Deli"->"New Delhi",
"नयाँ दिल्ली"->"New Delhi",
"Novi Deli"->"New Delhi",
"Nòva Delhi"->"New Delhi",
"Нью-Дели"->"New Delhi",
"ਨਵੀਂ ਦਿੱਲੀ"->"New Delhi",
"Nowe Delhi"->"New Delhi",
"نئی دلی"->"New Delhi",
"Nova Deli"->"New Delhi",

"Musuq Dilhi"->"New Delhi",
"Нью-Дели"->"New Delhi",
"नवदेहली"->"New Delhi",
"Сања Дели"->"New Delhi",
"Nova Delhi"->"New Delhi",
"නව දිල්ලිය"->"New Delhi",
"Naí Dillí"->"New Delhi",
"Њу Делхи"->"New Delhi",
"புது தில்லி"->"New Delhi",
"క్రొత్త ఢిల్లీ"->"New Delhi",
"Ню-Дели"->"New Delhi",
"นิวเดลี"->"New Delhi",
"Bagong Delhi"->"New Delhi",
"Yeni Delhi"->"New Delhi",
"Yéngi Déhli"->"New Delhi",
"Нью-Делі"->"New Delhi",
"نئی دہلی"->"New Delhi",
"Tân Delhi"->"New Delhi",
"ניו דעלי"->"New Delhi",
"新德里"->"New Delhi",
"Sin Delhi"->"New Delhi",
"新德里"->"New Delhi"

3. Data Overview

File sizes->

- ahmedabad_india.osm-> 23.8 MB
- nodes_csv-> 8.77 MB
- nodes_tags.csv-> 155 KB
- ways_csv-> 1.18 MB
- ways_nodes.csv-> 3.15 MB
- ways_tags.csv-> 833 KB
- ahmedabad.db-> 16.2MB

By applying queries I have few results described below

Number of nodes->

```
SQLite> SELECT COUNT (*) FROM nodes
```

Output: 109793

Number of ways->

```
SQLite> SELECT COUNT (*) FROM ways
```

Output: 20268

Number of unique users->

```
SQLite> SELECT COUNT (DISTINCT (e.uid))
FROM (SELECT uid FROM nodes UNION ALL SELECT uid FROM ways) e;
```

Output: 410

Top contributing users->

```
SQLite> SELECT e.user, COUNT (*) as num
FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) e
GROUP BY e.user
ORDER BY num DESC
LIMIT 10;
```

Output:

usaikumar	41413
PlaneMad	12412
kranthikumar	10862
Jothirnadh	7384
himalay	6585
Ashok09	5658
Oberaffe	4551
ngarh	3932
vamshiN	3631
pvprasad	3207

Number of users contributing only once->

```
SQLite> SELECT COUNT(*)
FROM
(SELECT e.user, COUNT(*) as num
FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) e
GROUP BY e.user
HAVING num=1) u;
```

Output: 134

4. Additional Data Exploration

Common amenities->

```
SQLite> SELECT value, COUNT(*) as num
FROM nodes_tags
WHERE key='amenity'
GROUP BY value
ORDER BY num DESC
LIMIT 10;
```

Output: Restaurant - 88

Biggest religion:

```
SQLite> SELECT nodes_tags.value, COUNT(*) as num
FROM nodes_tags
JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='place_of_worship') i
ON nodes_tags.id=i.id
WHERE nodes_tags.key='religion'
GROUP BY nodes_tags.value
ORDER BY num DESC
LIMIT 1;
```

Output: Hindu- 6

Popular cuisines

```
SQLite> SELECT nodes_tags.value, COUNT(*) as num
FROM nodes_tags
JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='restaurant') i
```

```
ON nodes_tags.id=i.id WHERE nodes_tags.key='cuisine'
GROUP BY nodes_tags.value
ORDER BY num DESC;
```

Output: Indian -5

5. Conclusion

By Analyzing the Central Delhi map, I saw that it contains lots of street name in Hindi and English and somewhere also in other languages. I have also seen that there are few abbreviations which are used in the dataset. So I have corrected the data in other languages (leaving Hindi, as it should remain there for layman in India) to English, for consistency and also corrected the abbreviation. Other than that the overall data has fair amount of correct street names. Also since I have travelled many places in central Delhi, there are lots of tourist attraction which should be there in the map.

6. Additional Ideas

To begin with first and foremost we can make some rules or patterns to input data which users follow every time to input their data. This will also restrict users input in their native language, as shown above there are lots of data which is in different languages.

Secondly I think that since the data is so rich and since there are so many tourist place nearby central Delhi, the government should also contribute in the data so that tourists can get an idea about that.

Since there are so many branded shops, pubs and restaurant, to make the data the more informative we can ask the shop and pub owners to describe their surrounding region more appropriate and hence in some way increasing their visibility.

Benefits of these initiative: People can get clean and more informative about the city, making it possible either for any new person to find the place or it can help various logistics companies also to optimize the delivery root.

Anticipated problems : In my view there is not as such problem, it is just we have to bring common people together who are on daily basis going from these roots to update the information for a greater good.