# OpenStreetMap Data Wrangling Case Study

## Map Area

The map I chose was of Charlotte, NC, the closest major city to me, and the city I visit fairly often.

- https://www.openstreetmap.org/relation/177415

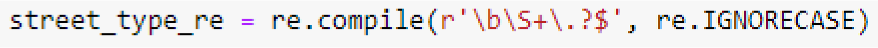
## Problems Encountered in the Data

I encountered a few problems in the data set; they are as follows:

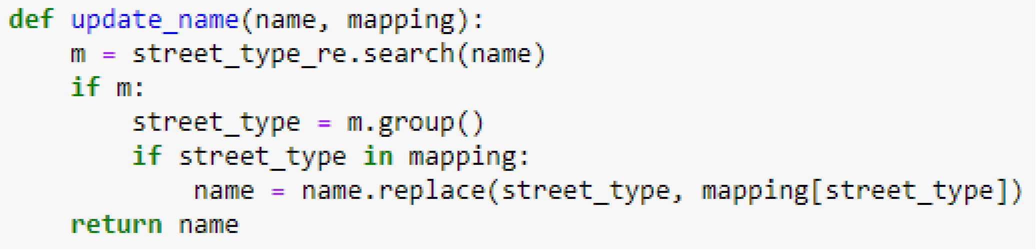
- Variance in street names and types
- Missing or redacted usernames
- Odd values in second level 'k' tags

### Variance in Street Names and Types

After auditing the data in python, it was clear that there was quite a lot of variance in street types. I used the code from the Udacity practice quizzes to standardize most of the street types.



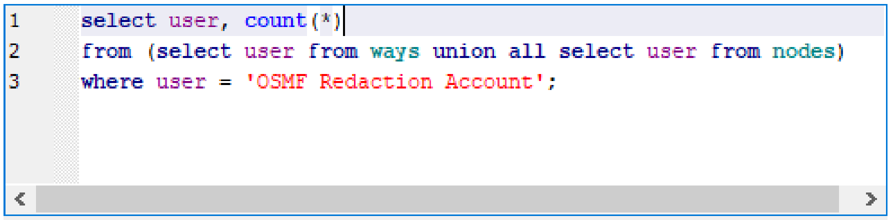
```
street_type_re = re.compile(r'\b\S+\.?$', re.IGNORECASE)
```



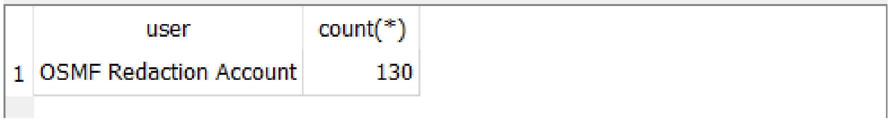
```
def update_name(name, mapping):
    m = street_type_re.search(name)
    if m:
        street_type = m.group()
        if street_type in mapping:
            name = name.replace(street_type, mapping[street_type])
    return name
```

### Missing or Redacted Usernames

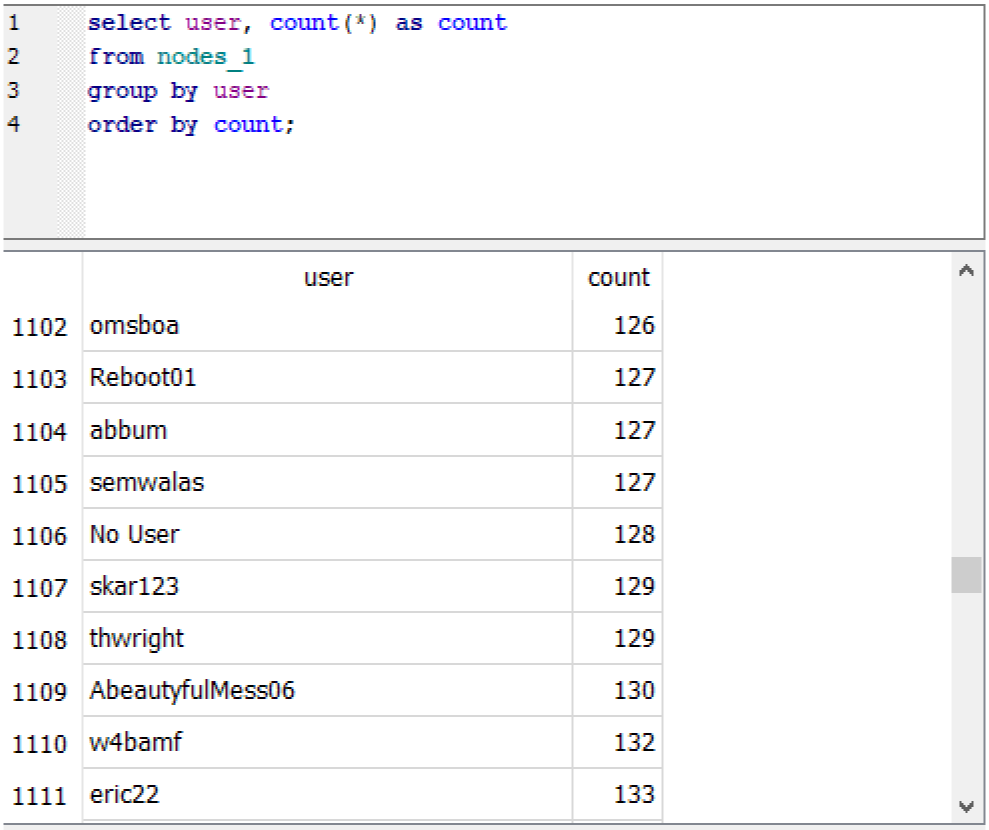
After porting the data into a SQL database and exploring it a bit, I found that there were many users' names who were labelled as 'OSMF Redaction Account' and although it's possible this was an actual user's name, I assumed that this was a mishap and replaced those names with a default value of 'No User'.



```
select user, count(*)
from (select user from ways union all select user from nodes)
where user = 'OSMF Redaction Account';
```



|   | user | count(*) |
|---|---|---|
| 1 | OSMF Redaction Account | 130 |

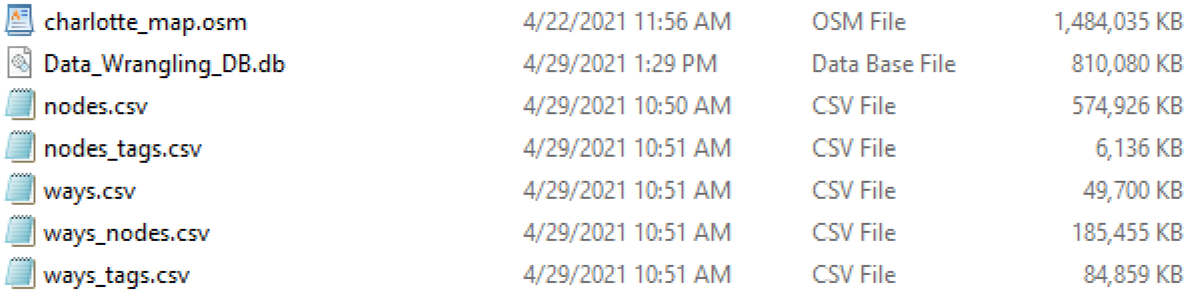


```
select user, count(*) as count
from nodes_1
group by user
order by count;
```

| | user | count |
|---|---|---|
| 1102 | omsboa | 126 |
| 1103 | Reboot01 | 127 |
| 1104 | abbum | 127 |
| 1105 | semwalas | 127 |
| 1106 | No User | 128 |
| 1107 | skar123 | 129 |
| 1108 | thwright | 129 |
| 1109 | AbeautyfulMess06 | 130 |
| 1110 | w4bamf | 132 |
| 1111 | eric22 | 133 |

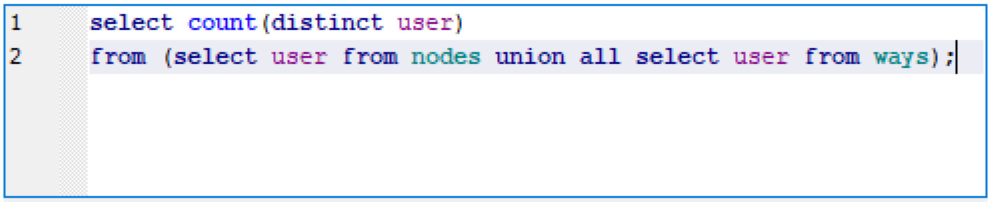
# Data Queries and Additional Thoughts

## File Sizes

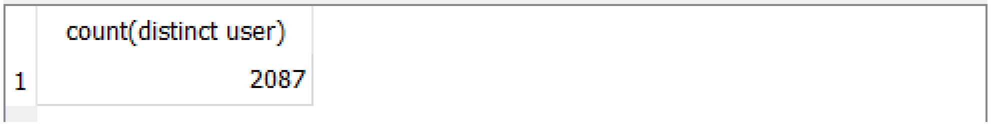


| | | | |
|---|---|---|---|
| charlotte_map.osm | 4/22/2021 11:56 AM | OSM File | 1,484,035 KB |
| Data_Wrangling_DB.db | 4/29/2021 1:29 PM | Data Base File | 810,080 KB |
| nodes.csv | 4/29/2021 10:50 AM | CSV File | 574,926 KB |
| nodes_tags.csv | 4/29/2021 10:51 AM | CSV File | 6,136 KB |
| ways.csv | 4/29/2021 10:51 AM | CSV File | 49,700 KB |
| ways_nodes.csv | 4/29/2021 10:51 AM | CSV File | 185,455 KB |
| ways_tags.csv | 4/29/2021 10:51 AM | CSV File | 84,859 KB |

## Number of Distinct Users

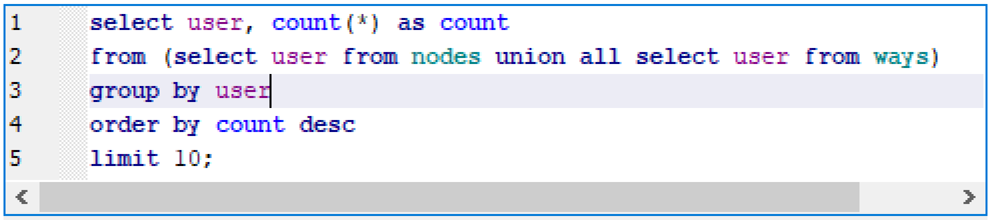


```sql
select count(distinct user)
from (select user from nodes union all select user from ways);
```

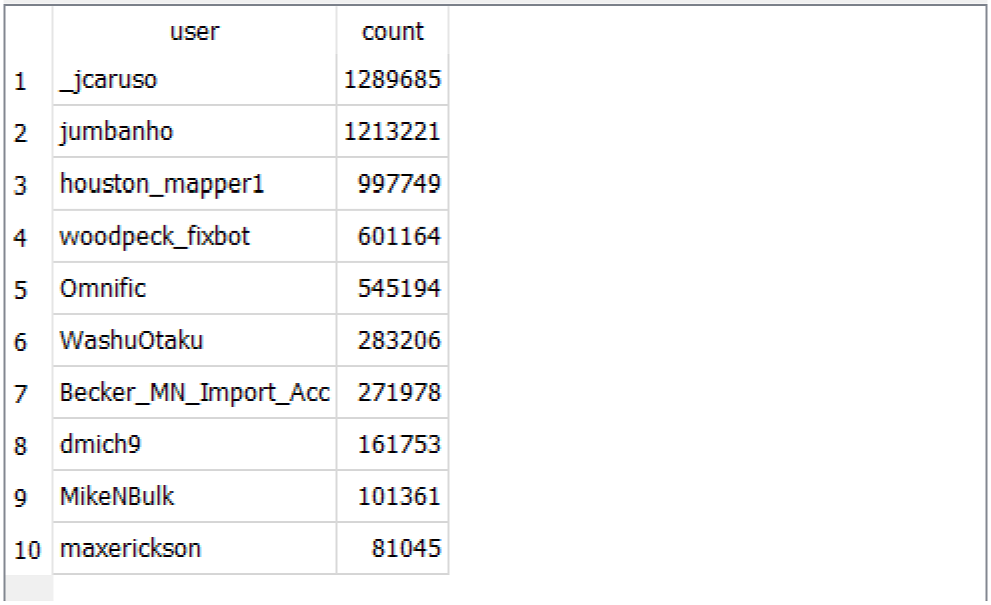


|   | count(distinct user) |
|---|---|
| 1 | 2087 |

## Top 10 Contributing Users

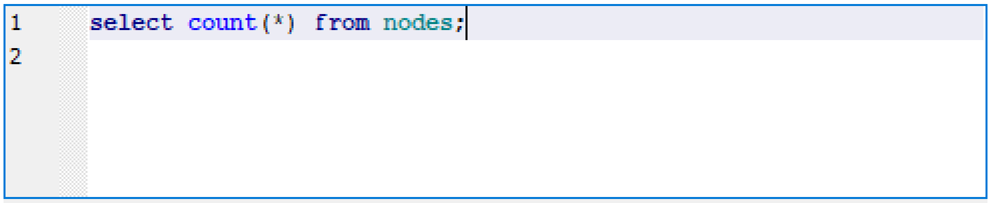


```sql
select user, count(*) as count
from (select user from nodes union all select user from ways)
group by user
order by count desc
limit 10;
```

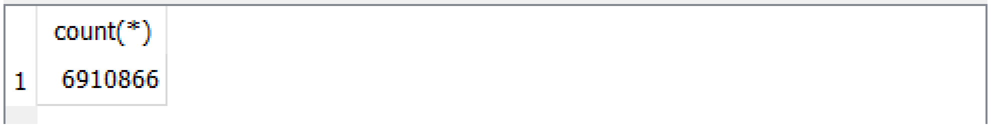


|   | user | count |
|---|---|---|
| 1 | _jcaruso | 1289685 |
| 2 | jumbanho | 1213221 |
| 3 | houston_mapper1 | 997749 |
| 4 | woodpeck_fixbot | 601164 |
| 5 | Omnific | 545194 |
| 6 | WashuOtaku | 283206 |
| 7 | Becker_MN_Import_Acc | 271978 |
| 8 | dmich9 | 161753 |
| 9 | MikeNBulk | 101361 |
| 10 | maxerickson | 81045 |

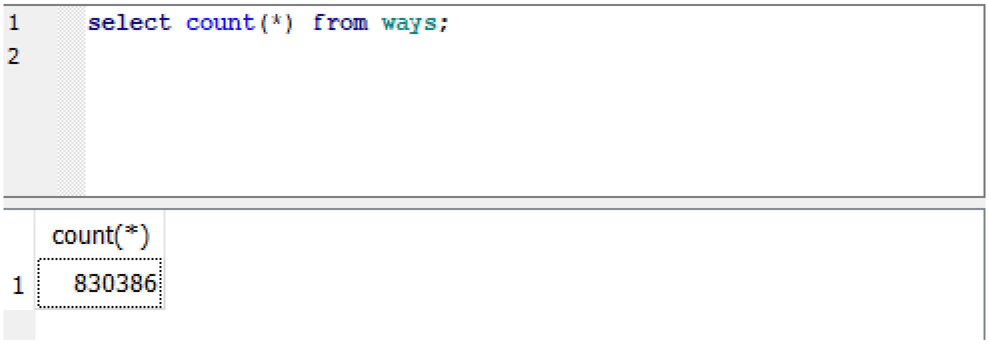
## Number of Nodes, Ways, and Both



```sql
select count(*) from nodes;
```

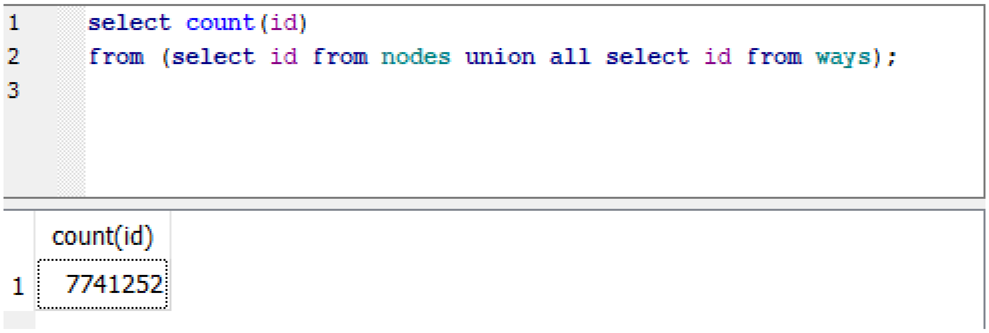


|   | count(*) |
|---|---|
| 1 | 6910866 |



```sql
select count(*) from ways;
```

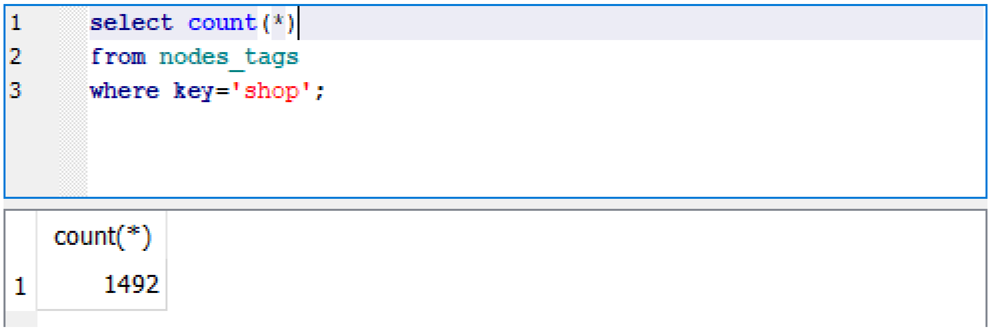
```
count(*)
830386
```



```sql
select count(id)
from (select id from nodes union all select id from ways);
```

```
count(id)
7741252
```

### Number of Shops

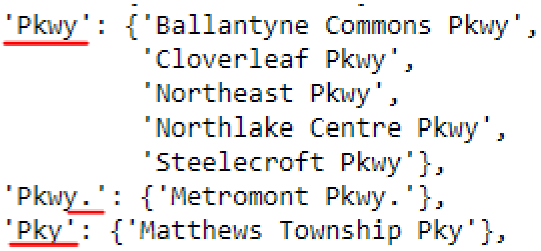


```sql
select count(*)
from nodes_tags
where key='shop';
```

```
count(*)
1492
```

## Additional Thoughts

One of the major problems with the data from OpenStreetMap is the variance in street names and abbreviations of street types. Because the data is user entered and open for anyone to submit data, nothing is standardized. There can even be multiple different abbreviations of the same street submitted by different users. A small example is this:

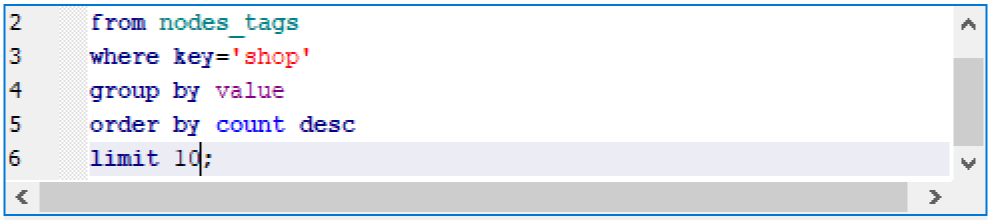


```
'Pkwy': {'Ballantyne Commons Pkwy',
         'Cloverleaf Pkwy',
         'Northeast Pkwy',
         'Northlake Centre Pkwy',
         'Steelecroft Pkwy'},
'Pkwy.': {'Metromont Pkwy.'},
'Pky': {'Matthews Township Pky'},
```

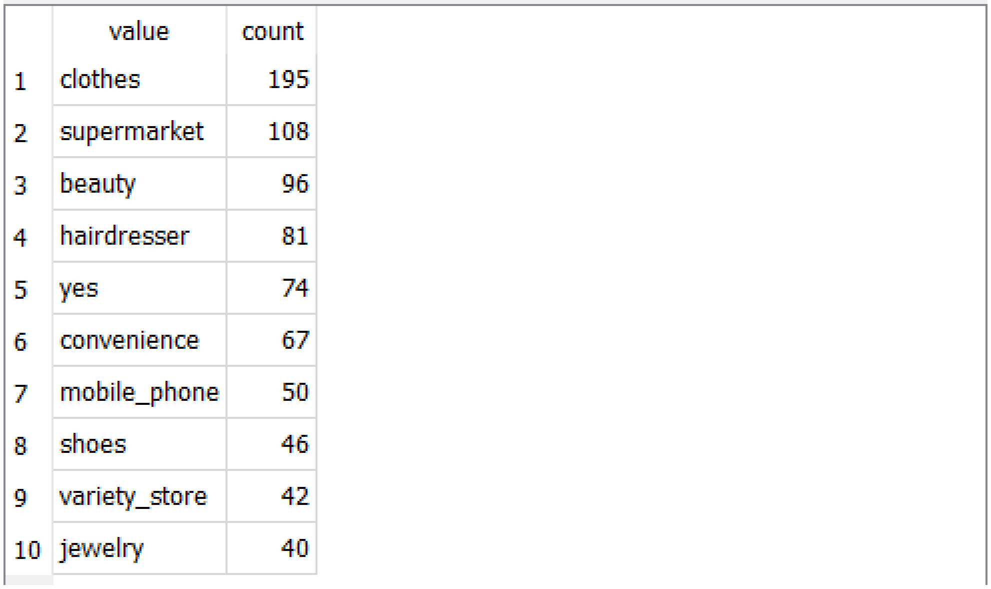
There were three different abbreviations for the street type Parkway; Pkwy, Pkwy., and Pky. A script could be made and implemented on the website to run over data that is entered and standardize at least the most common street types. Benefits of this are that cleaning the data for anyone who wants to use it would be easier and quicker if most street types were corrected upon entering the data into OpenStreetMap. The difficulty in doing this however is that so many different abbreviations exist for streets, and OpenStreetMap has data from all over the world meaning streets are in so many different languages, it would take a lot of time and effort for one or a few people to even write the scripts for English speaking countries, let alone every other country the site has data for.

# Additional Statistics

### Most Common Shop Types



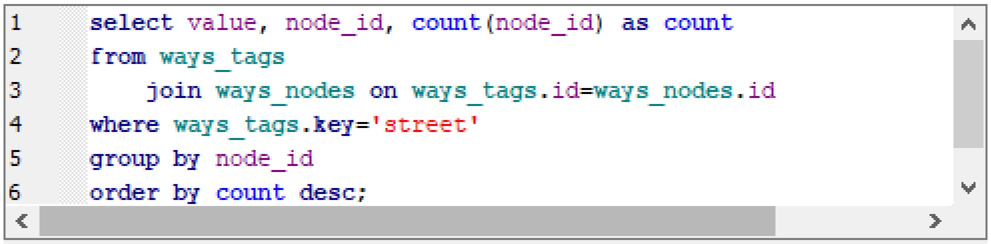
```
from nodes_tags
where key='shop'
group by value
order by count desc
limit 10;
```



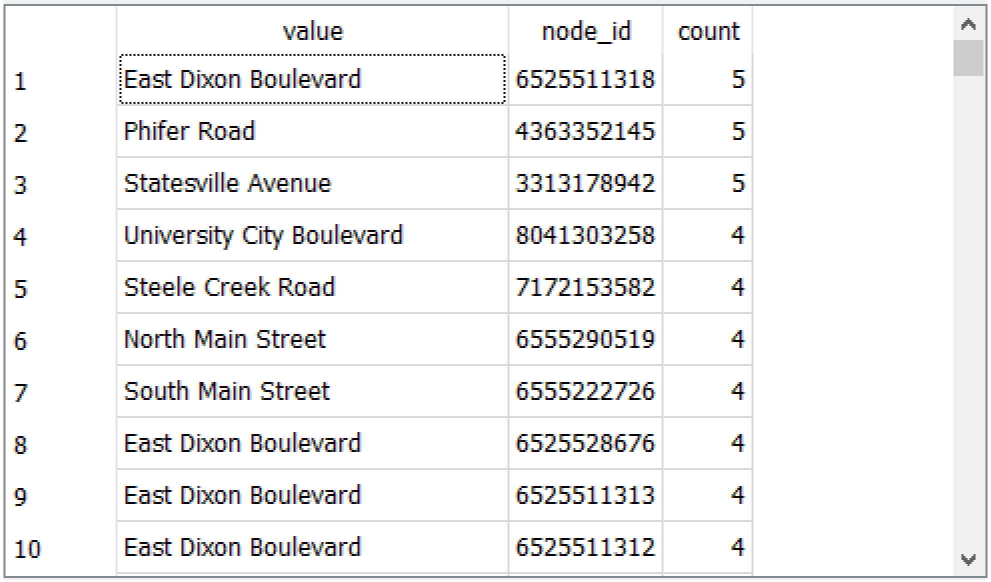
| | value | count |
|---|---|---|
| 1 | clothes | 195 |
| 2 | supermarket | 108 |
| 3 | beauty | 96 |
| 4 | hairdresser | 81 |
| 5 | yes | 74 |
| 6 | convenience | 67 |
| 7 | mobile_phone | 50 |
| 8 | shoes | 46 |
| 9 | variety_store | 42 |
| 10 | jewelry | 40 |

### Multiple Copies of Streets

Although this one is a statistic due to errors of a kind, I still felt it should be included here, as it was interesting and supports my script addition. This statistic shows that there are multiple instances of the same node_id appearing in the same street, in different way tags. This means that there are multiple redundant copies of the same nodes under the streets.



```
select value, node_id, count(node_id) as count
from ways_tags
    join ways_nodes on ways_tags.id=ways_nodes.id
where ways_tags.key='street'
group by node_id
order by count desc;
```



| | value | node_id | count |
|---|---|---|---|
| 1 | East Dixon Boulevard | 6525511318 | 5 |
| 2 | Phifer Road | 4363352145 | 5 |
| 3 | Statesville Avenue | 3313178942 | 5 |
| 4 | University City Boulevard | 8041303258 | 4 |
| 5 | Steele Creek Road | 7172153582 | 4 |
| 6 | North Main Street | 6555290519 | 4 |
| 7 | South Main Street | 6555222726 | 4 |
| 8 | East Dixon Boulevard | 6525528676 | 4 |
| 9 | East Dixon Boulevard | 6525511313 | 4 |
| 10 | East Dixon Boulevard | 6525511312 | 4 |

## Conclusion

In conclusion, it is clear that the data entered for Charlotte, NC is varied and incomplete. For the purposes of this project I believe the data has been cleaned thoroughly enough. However, it is clear that the data needs to be cleaned and standardized more thoroughly when it is entered into OpenStreetMap. It's interesting to see that there is so much redundancy in the nodes included in the streets. With scripts to standardize the street types and prevent multiple copies of the same street, the data on OpenStreetMap.org could be cleaned and organized much better and more intuitively, making it easier to work with and reducing the file size.