Map Area

Seattle WA, United States

https://www.openstreetmap.org/export#map=10/47.6258/-122.3252

Problems Encountered in the Map

I have downloaded a small sample size of the Seattle area and run it, I noticed the following five main problems:

- Misleading abbreviation for street names (“142nd Ave E”, “S River Rd”)
- Inconsistent postal codes (“98057-4040″ , “98020”)

Correcting Street Names
Once the data was imported to SQL, some basic querying revealed street name abbreviations and postal code inconsistencies. To deal with correcting street names, I opted not use regular expressions, and instead iterated over each word in an address, correcting them to their respective mappings in audit.py using the following function:

```
def update_name(name, mapping):


    for key,value in mapping.iteritems():
        key_type_re = re.compile(r'\S+\.?$', re.IGNORECASE)
        m = key_type_re.search(name)
        if m:
            key_type = m.group()
        if key_type in mapping.keys():
            # substitute the street_type for its clean version in 'name'
            name = re.sub( key_type, mapping[key_type], name)


        return name
```

Postal Codes
Postal code strings are mostly 5 digits with 980 as starting number, but some are4-digit zip code extensions following a hyphen (“98057-4040”). After standardizing inconsistent postal codes, some altogether postal codes showed up in the following manner:

```
query = "SELECT tags.value, COUNT(*) as count FROM (SELECT * FROM nodes_tags UNION ALL \
        SELECT * FROM ways_tags) tags \
        WHERE tags.key='postcode'\
        GROUP BY tags.value\
        ORDER BY count DESC;"
c.execute(query)
rows = c.fetchall()
```

Here are the top ten results, beginning with the highest count:

```
print(rows)
```

```
[(u'98178', 249), (u'98118', 114), (u'98057', 23), (u'98371', 19), (u'98092', 13), (u'98042', 11), (u'98032', 10), (u'98001', 4), (u'98030', 4), (u'98354', 4), (u'98010', 2), (u'98027', 2), (u'98038', 2), (u'98055', 2), (u'98058', 2), (u'98059', 2), (u'98372', 2), (u'98391', 2), (u'90092', 1), (u'98003', 1), (u'98032-1762', 1), (u'98057-4040', 1), (u'98104', 1), (u'98108', 1), (u'98168', 1), (u'98188', 1), (u'98198', 1), (u'98321', 1), (u'98373', 1), (u'98374', 1)]
```

Overview of Data

File Sizes

```
Seattlemap.osm ......... 129 MB
sample.osm ............. 13 MB
seattle.db ............. 7 MB
nodes.csv .............. 4 MB
nodes_tags.csv ......... 267 KB
ways.csv ............... 437 KB
ways_tags.csv .......... 974 KB
ways_nodes.cv .......... 1.7 MB
```

Number of Ways

```
query = "select count(*) from ways"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(7266,)]
```

Number of Unique Users

```
query = "select count(distinct(e.uid)) from(select uid from nodes UNION ALL select uid from ways) e"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(436,)]
```

Top 10 Contributing Users

```
query = "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;"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(u'Omnific', 19989), (u'Grauer Elefant', 4687), (u'Amoebabadass', 4468), (u'STBrenden', 3977), (u'Glassman', 3680), (u'csytsm
a', 2836), (u'woodpeck_fixbot', 2403), (u'Geodesy99', 2245), (u'Skybunny', 1766), (u'zephyr', 1414)]
```

Number of Users Appearing only once (having 1 post)

```
query = "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;"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(107,)]
```

Additional Ideas

Top 10 Appearing Amenities

```
query = "SELECT value, COUNT(*) as num FROM nodes_tags\
       WHERE key='amenity'\
       GROUP BY value\
       ORDER BY num DESC\
       LIMIT 10;"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(u'restaurant', 42), (u'bench', 38), (u'fast_food', 34), (u'waste_basket', 20), (u'cafe', 19), (u'bank', 15), (u'doctors', 1
5), (u'fuel', 13), (u'dentist', 12), (u'school', 12)]
```

Biggest Religion

```
query = "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;"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(u'christian', 6)]
```

Most Popular Cuisines

```
query = "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;"
c.execute(query)
rows = c.fetchall()
print(rows)
```

```
[(u'pizza', 7), (u'japanese', 4), (u'chinese', 3), (u'asian', 2), (u'burger', 2), (u'mexican', 2), (u'thai', 2), (u'american;savory_pancakes;pancake;breakfast', 1), (u'barbecue', 1), (u'burger;asian', 1), (u'indian', 1), (u'italian', 1), (u'pancake;breakfast', 1), (u'vietnamese', 1)]
```