CIS5200 Term Project Tutorial

Authors: Aakanksha Tasgaonkar, Amogh Mahesh, Rupa Mahendran, Sriganesh Baskaran

Instructor: Jongwook Woo

Date: 04/12/2018

Lab Tutorial

Aakanksha Tasgaonkar (atasgao@calstatela.edu)

Amogh Mahesh (amahesh3@calstatela.edu)

Rupa Mahendran (rmahend@calstatela.edu)

Sriganesh Baskaran (sbaskar2@calstatela.edu)

04/12/2018

Data Analysis using Spark (NYC-Parking Ticket)

Objectives

In this hands-on lab, you will learn how to:

- Uploaded the dataset for New York Parking system
- Data cleaning using Hive
- HiveQL commands to perform the analysis
- Visualization in Power BI, Tableau and 3D map

Platform Spec

- Oracle Big Data Compute Edition
- CPU Speed: 2.195 GHz
- # of CPU cores: 4 cores, 1 Socket
- # of nodes: 5 nodes
- Total Memory Size: 32GB

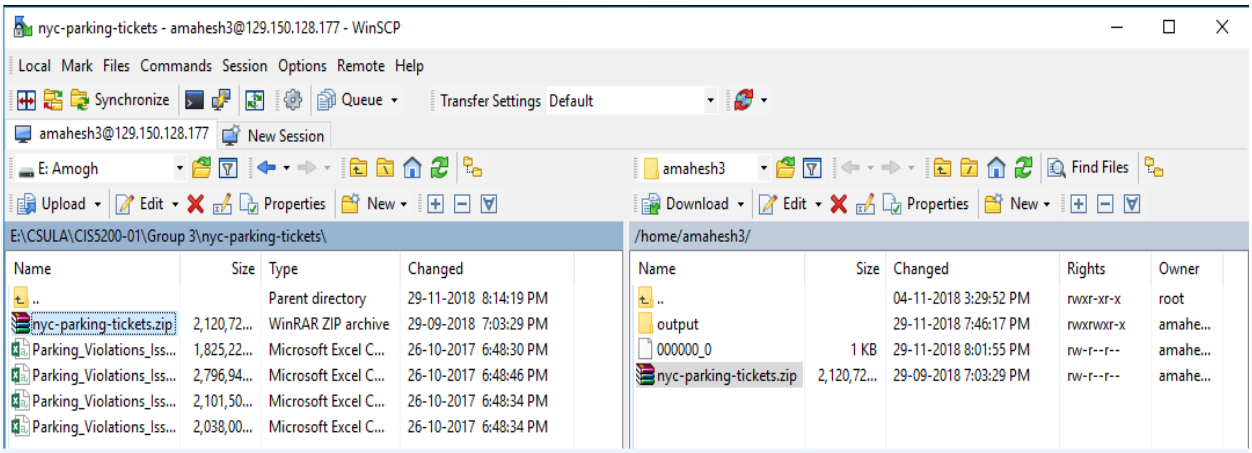
INTRODUCTION

The NYC Department of Finance collects data on every parking ticket issued in NYC (~10M per year!). The Department of Finance is responsible for collecting and processing payments for all parking tickets and camera violations. This dataset gives an insight on ticket resolution and to guide policymakers. This project will illustrate the usage of **Hadoop, MapReduce, HiveQL** to generate data, Microsoft PowerBI and 3D map to visualize data.

Step 1: Data Upload to server

Server IP: 129.150.129.67

- Download the dataset from the url: **https://www.kaggle.com/new-york-city/nyc-parking-tickets**
- Use the software WinSCP to upload the dataset **nyc-parking-tickets.zip** of size 2GB [8GB unzipped] from local system to oracle server.



- Use the command **unzip nyc-parking-tickets.zip** to extract the files as a csv format.

```
-bash-4.1$ ls
000000_0  nyc-parking-tickets.zip  output
-bash-4.1$ unzip nyc-parking-tickets.zip
Archive:  nyc-parking-tickets.zip
  inflating: Parking_Violations_Issued_-_Fiscal_Year_2014__August_2013___June_2014_.csv
  inflating: Parking_Violations_Issued_-_Fiscal_Year_2017.csv
  inflating: Parking_Violations_Issued_-_Fiscal_Year_2016.csv
  inflating: Parking_Violations_Issued_-_Fiscal_Year_2015.csv
-bash-4.1$ ls
000000_0                Parking_Violations_Issued_-_Fiscal_Year_2014__August_2013___June_2014_.csv  Parking_Violations_Issued_-_Fiscal_Year_2017.csv
nyc-parking-tickets.zip  Parking_Violations_Issued_-_Fiscal_Year_2015.csv
output                  Parking_Violations_Issued_-_Fiscal_Year_2016.csv
```

Step 2: Uploading data to HDFS

- Create folders using **mkdir** command, Change the permission for the file using below command to get full access to the file.

```
hdfs dfs -mkdir Group3/master2014
hdfs dfs -chmod -R o+w Group3/master2014
```

Note: replace **2014** with other years.

- Use **hdfs dfs -put** command to upload the data from the Linux server to HDFS

```
hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2014.csv
hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2015.csv
hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2016.csv
hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2017.csv
```

```
-bash-4.1$ hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2014.csv
-bash-4.1$ hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2015.csv
-bash-4.1$ hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2016.csv
-bash-4.1$ hdfs dfs -put Parking_Violations_Issued_-_Fiscal_Year_2017.csv
```

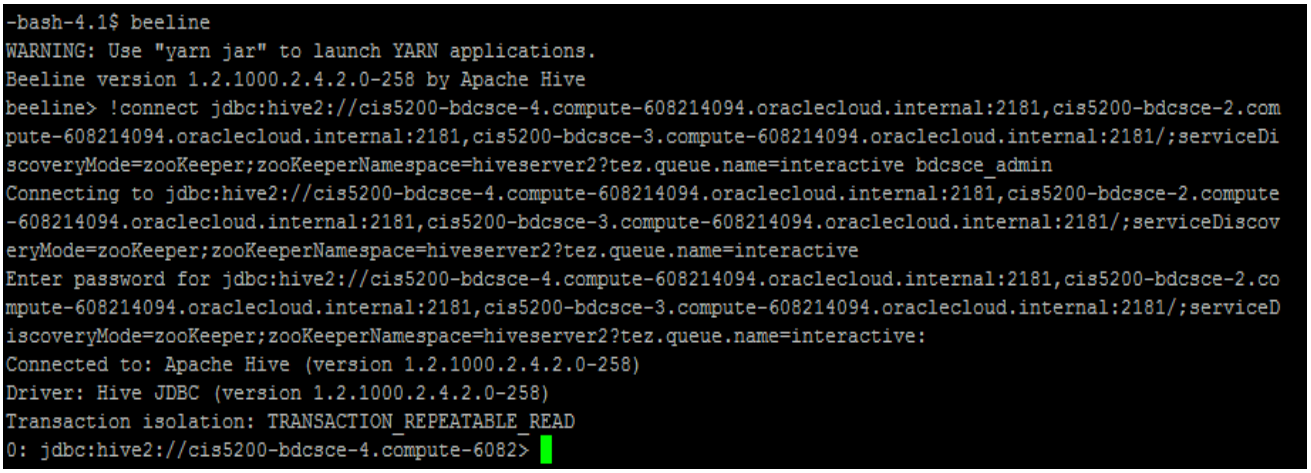
- Use **hdfs dfs -mv** to move file from hdfs parent folder to created directory.

```
-bash-4.1$ hdfs dfs -mkdir Group3/master2014
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2014/
-bash-4.1$ hdfs dfs -mv Parking_Violations_Issued_-_Fiscal_Year_2014.csv Group3/master2014/
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2014/
-bash-4.1$ hdfs dfs -mkdir Group3/master2015
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2015/
-bash-4.1$ hdfs dfs -mv Parking_Violations_Issued_-_Fiscal_Year_2015.csv Group3/master2015/
-bash-4.1$ hdfs dfs -mkdir Group3/master2016
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2016/
-bash-4.1$ hdfs dfs -mv Parking_Violations_Issued_-_Fiscal_Year_2016.csv Group3/master2016/
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2016/
-bash-4.1$ hdfs dfs -mkdir Group3/master2017
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2017/
-bash-4.1$ hdfs dfs -mv Parking_Violations_Issued_-_Fiscal_Year_2017.csv Group3/master2017/
-bash-4.1$ hdfs dfs -chmod -R o+w Group3/master2017/
```

Step 3: Connecting server to HIVE

- Use the command **beeline** to connect with to Hive

!connect jdbc:hive2://cis5200-bdcsce-4.compute-08214094.oraclecloud.internal:2181,cis5200-bdcsce-2.compute-608214094.oraclecloud.internal:2181,cis5200-bdcsce-3.compute-608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2?tez.queue.name=interactive bdcsce_admin



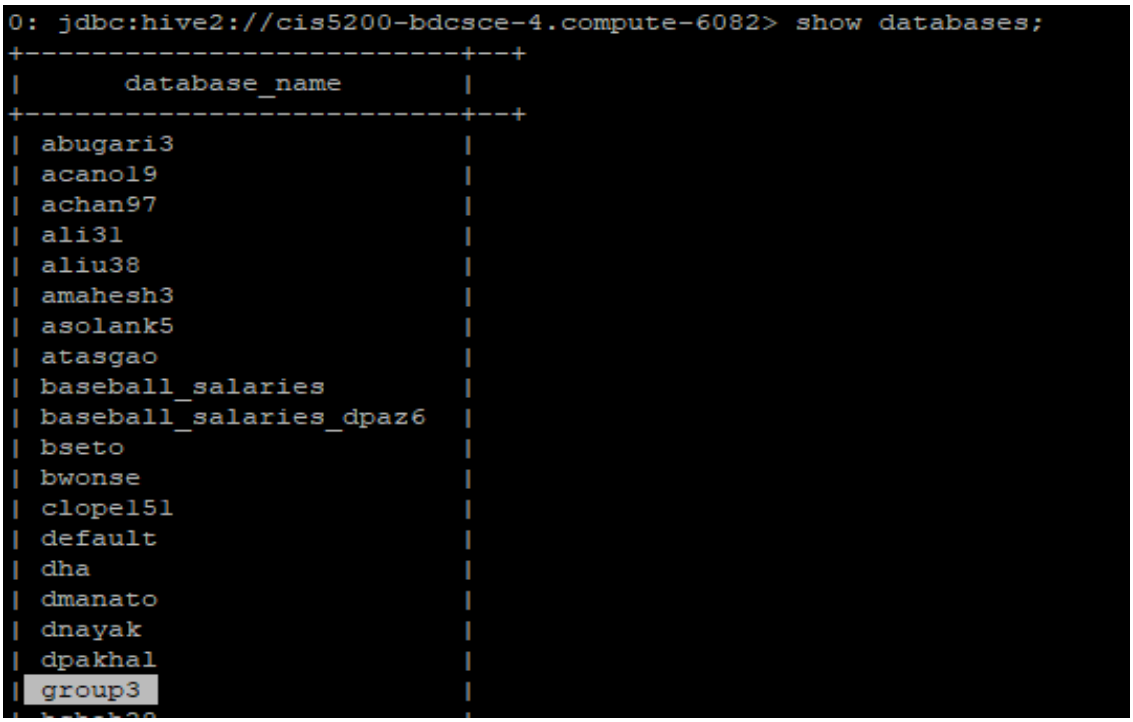
Step 4: Creating Database

- Use **create database** to create a database named group3.

```
create database group3;
```

- To view the newly created database use.

```
Show databases;
```



Step 5: Creating Tables in Database

- To **Create External table**, use below command,

```
CREATE EXTERNAL TABLE IF NOT EXISTS NYC_parking2014(Summons_Number
      BIGINT,Plate_ID STRING,Registration_State STRING,Plate_Type
      STRING,Issue_Date STRING,Violation_Code BIGINT,Vehicle_Body_Type
      STRING,Vehicle_Make STRING,Issuing_Agency STRING,Street_Code1
      BIGINT,Street_Code2 BIGINT,Street_Code3
      BIGINT,Vehicle_Expiration_Date BIGINT,Violation_Location
      STRING,Violation_Precinct BIGINT,Issuer_Precinct BIGINT,Issuer_Code
      BIGINT,Issuer_Command STRING,Issuer_Squad STRING,Violation_Time
      STRING,Time_First_Observed STRING,Violation_County
      STRING,Violation_In_Front_Of_Or_Opposite STRING,House_number
      BIGINT,Street_Name STRING,Intersecting_Street
      STRING,Date_First_Observed BIGINT,Law_Section BIGINT,Sub_Division
      STRING,Violation_Legal_Code STRING,Days_Parking_In_Effect
      STRING,From_Hours_In_Effect STRING,To_Hours_In_Effect
      STRING,Vehicle_Color STRING,Unregistered_Vehicle
      STRING,Vehicle_Year BIGINT,Meter_number BIGINT,Feet_From_Curb
      BIGINT,Violation_Post_Code STRING,Violation_Description
      STRING,No_Standing_or_Stopping_Violation STRING,Hydrant_Violation
      STRING,Double_Parking_Violation STRING,Latitude BIGINT,Longitude
      BIGINT,Community_Board     BIGINT,Community_Council
      BIGINT,Census_Tract BIGINT,BIN BIGINT,BBL      BIGINT,NTA STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/user/amahesh3/Group3/master2014'
TBLPROPERTIES ('skip.header.line.count'='1');
```

NOTE: replace **2014** with other years (2015,2016,2017) and create tables.

Step 6: Dataset cleaning in Hive

As the New york city parking ticket dataset has more number of uwanted data like blank cells and irrelevent data it needs to be filter out and saved as New file Clean_nyc2014, Clean_nyc2015, Clean_nyc2016, Clean_nyc2017 in the path /user/amahesh3/Group3/

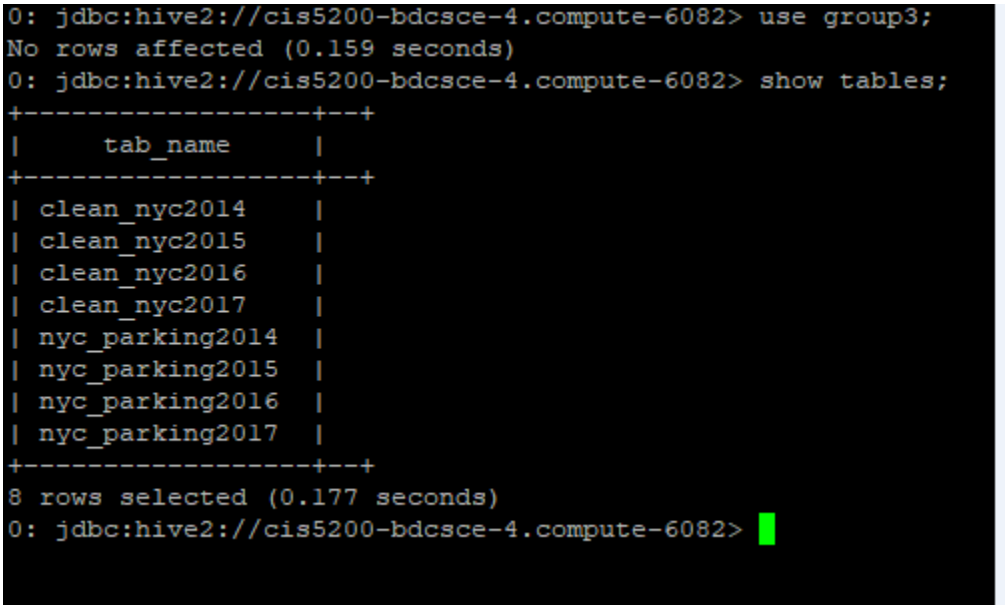
- Use the below command to clean the dataset,

```
CREATE TABLE clean_nyc2014
 ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
 STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014'
 AS
SELECT
      summons_number,plate_id,registration_state,plate_type,issue_date
      ,violation_code,vehicle_body_type,vehicle_make,issuing_agency,st
      reet_code1,street_code2,street_code3,vehicle_expiration_date,iss
      uer_code,issuer_command,violation_time,time_first_observed,viola
      tion_county,violation_in_front_of_or_opposite,house_number,stree
      t_name,date_first_observed,vehicle_color,vehicle_year,meter_numb
      er,violation_description from nyc_parking2014
where
      registration_state != 99 OR plate_id != 999 OR vehicle_make != '
      ' OR street_code1 != 0 OR street_code2 != 0 OR street_code3 != 0
      OR issuer_code != 0 OR time_first_observed != ' ' OR
      violation_county != ' ' OR violation_in_front_of_or_opposite !=
      ' ' OR house_number != ' ' OR vehicle_color != ' ' OR
      vehicle_year != 0 OR meter_number != ' ' OR
      violation_description != ' ';
```

NOTE: replace **2014** with other years (2015,2016,2017) and create tables.

- Use the command to view all the tables in the database,

  Show tables;

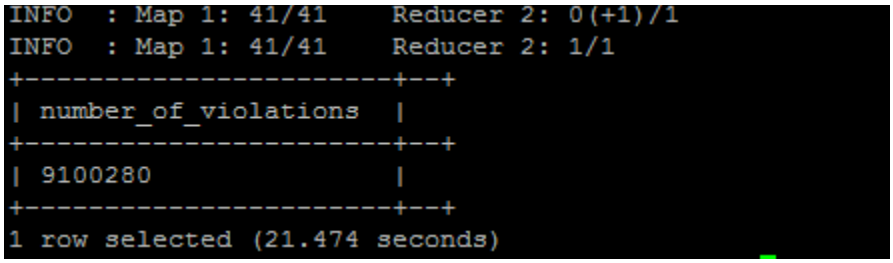


Step 7: Creating Hive Queries to Analyze Data

The following Hive queries analyses the New York city parking tickets for the following year **2014.**

1. **Query to analyse the Total number of Violated Parking tickets which has been recorded during in the year 2014.**

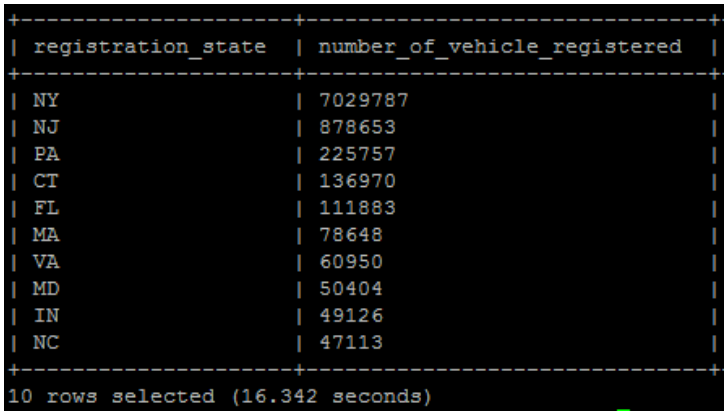
```
CREATE TABLE tickets_count_2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
/user/amahesh3/Group3/master2014/clean_nyc2014/tickets_count_2014'
AS
select count(summons_number) as number_of_violations from clean_nyc2014;
```



```
INFO  : Map 1: 41/41    Reducer 2: 0(+1)/1
INFO  : Map 1: 41/41    Reducer 2: 1/1
+-----------------------+--+
| number_of_violations  |
+-----------------------+--+
| 9100280               |
+-----------------------+--+
1 row selected (21.474 seconds)
```

2. **Query to find out top 10 registration state of the vehicle which has done Parking violation NYC in 2014**.

```
CREATE TABLE vehicle_registered_state2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/vehicle_registered_state2014'
AS
select registration_state, count(registration_state) as number_of_vehicle_registered from
clean_nyc2014 group by registration_state order by number_of_vehicle_registered desc
limit 10 ;
```



```
+--------------------+------------------------------+
| registration_state | number_of_vehicle_registered |
+--------------------+------------------------------+
| NY                 | 7029787                      |
| NJ                 | 878653                       |
| PA                 | 225757                       |
| CT                 | 136970                       |
| FL                 | 111883                       |
| MA                 | 78648                        |
| VA                 | 60950                        |
| MD                 | 50404                        |
| IN                 | 49126                        |
| NC                 | 47113                        |
+--------------------+------------------------------+
10 rows selected (16.342 seconds)
```

3. **Query to find out top 10 violations in NYC based on the vehicle body types in the year 2014.**

```
CREATE TABLE violations_vehicle_type2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/violations_vehicle_type2014'
AS
select vehicle_body_type, count(vehicle_body_type) as number_of_vehicles from
clean_nyc2014 where vehicle_body_type != "" group by vehicle_body_type order by
number_of_vehicles desc limit 10;
```

```
0: jdbc:hive2://cis5200-bdcsce-4.compute-6082> select * from violations_vehicle_type2014;
+-----------------------------------------------+-------------------------------------------------+--+
| violations_vehicle_type2014.vehicle_body_type  | violations_vehicle_type2014.number_of_vehicles  |
+-----------------------------------------------+-------------------------------------------------+--+
| SUBN                                          | 2797673                                         |
| 4DSD                                          | 2513564                                         |
| VAN                                           | 1384386                                         |
| DELV                                          | 730517                                          |
| SDN                                           | 441801                                          |
| 2DSD                                          | 255200                                          |
| PICK                                          | 226119                                          |
| UTIL                                          | 78454                                           |
| REFG                                          | 64904                                           |
| TRAC                                          | 62699                                           |
+-----------------------------------------------+-------------------------------------------------+--+
10 rows selected (0.076 seconds)
```

4. **Query to find out top 10 vehicle which has violated the parking law of New York city frequently in the following years,**

```
CREATE TABLE tickets_for_plate_id2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/tickets_for_plate_id2014'
AS
select plate_id , vehicle_body_type, vehicle_make, count(plate_id) as tickets_received
from clean_nyc2015 where plate_id != "BLANKPLATE" group by plate_id ,
vehicle_make, vehicle_body_type  order by tickets_received desc limit 10 ;
```

plate_id	vehicle_body_type	vehicle_make	tickets_received
62901JM	DELV	INTER	958
92979JE	DELV	INTER	864
47603MD	DELV	PETER	846
75225JW	DELV	FRUEH	842
62627JM	DELV	INTER	768
68092JZ	DELV	HINO	767
17442JE	DELV	INTER	760
63098JM	VAN	GMC	735
17744MD	DELV	PETER	709
49781MA	DELV	KENWO	676

10 rows selected (34.641 seconds)

5. Query to find out Number of tickets recorded during public holidays in NYC

```
CREATE TABLE tickets_on_public_holidays2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/tickets_on_public_holidays2014'
AS
select issue_date, count(issue_date) as number_of_violations from clean_nyc2014 where
issue_date in ( "12/25/2014" , "01/01/2014", "07/04/2014", "09/03/2014", "11/28/2014")
group by issue_date limit 10;
```

issue_date	number_of_violations
11/28/2014	802
01/01/2014	5114
12/25/2014	254
09/03/2014	211
07/04/2014	1050

5 rows selected (22.79 seconds)

6. **Query to find out top 5 dates which got recorded with maximum number of tickets in year 2014**.

```
CREATE TABLE date_high_tickets2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/date_high_tickets2014'
AS
select issue_date, count(issue_date) as number_of_violations from clean_nyc2014  group by
issue_date order by number_of_violations desc limit 5;
```

```
+-------------+-----------------------+-
| issue_date  | number_of_violations  |
+-------------+-----------------------+-
| 11/29/2013  | 46023                 |
| 10/03/2013  | 41359                 |
| 10/08/2013  | 41103                 |
| 10/01/2013  | 40664                 |
| 03/06/2014  | 40639                 |
+-------------+-----------------------+-
5 rows selected (23.221 seconds)
```

7. **Query to find out top 5 vehicle brand received highest number of tickets in 2014.**

```
CREATE TABLE violation_wrt_vehicle_make2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/violation_wrt_vehicle_make2014'
AS
select vehicle_make, count (vehicle_make ) as number_of_violations from clean_nyc2014 group
by vehicle_make order by number_of_violations desc limit 5;
```

```
+---------------+-----------------------+--+
| vehicle_make  | number_of_violations  |
+---------------+-----------------------+--+
| FORD          | 1248618               |
| TOYOT         | 878781                |
| HONDA         | 810453                |
| CHEVR         | 743552                |
| NISSA         | 655271                |
+---------------+-----------------------+--+
5 rows selected (17.51 seconds)
```

8. Query to find out which street in New York city has got the greatest number of tickets.

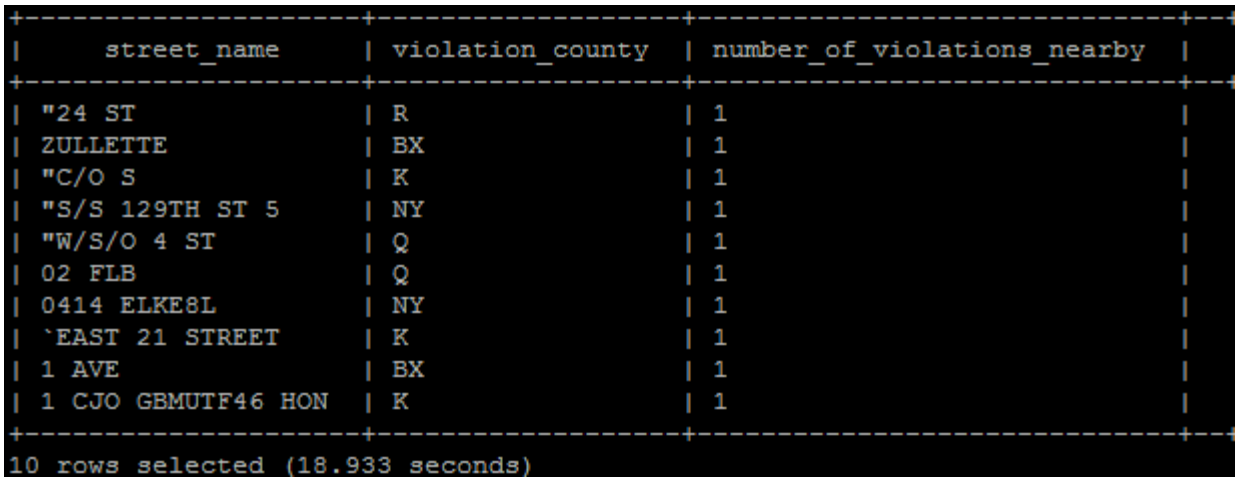
```
CREATE TABLE most_violated_location2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/most_violated_location2014'
AS
select
violation_in_front_of_or_opposite,house_number,street_name,violation_county,count(house
_number) as number_of_violations_nearby from clean_nyc2014 group by
violation_in_front_of_or_opposite,house_number,street_name,violation_county order by
number_of_violations_nearby desc limit 10;
```

violation_in_front_of_or_opposite	house_number	street_name	violation_county	number_of_violations_nearby
F	2655	Richmond Ave	R	2091
O	365	Bridge St	K	1932
F	448	W 16th St	NY	1728
F	16	E 58th St	NY	1515
F	55	E 59th St	NY	1421
F	99	Warren St	NY	1369
O	1330	1st Ave	NY	1357
F	1546	2nd Ave	NY	1353
F	20	Pine St	NY	1343
F	12	E 49th St	NY	1340

10 rows selected (25.514 seconds)

9. Street which has a least number of parking tickets recorded in 2014.

```
CREATE TABLE least_violated_location2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/least_violated_location2014'
AS
select street_name,violation_county,count(street_name) as number_of_violations_nearby
from clean_nyc2014 where street_name != '' AND violation_county != '' group by
street_name,violation_county order by number_of_violations_nearby asc limit 10;
```

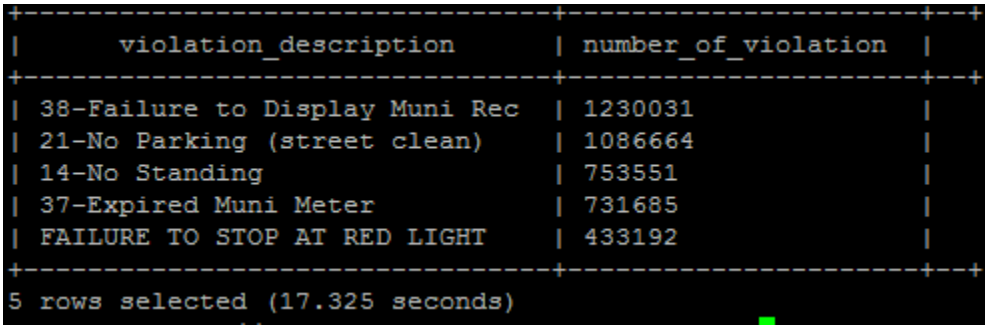


street_name	violation_county	number_of_violations_nearby
"24 ST	R	1
ZULLETTE	BX	1
"C/O S	K	1
"S/S 129TH ST 5	NY	1
"W/S/O 4 ST	Q	1
02 FLB	Q	1
0414 ELKE8L	NY	1
`EAST 21 STREET	K	1
1 AVE	BX	1
1 CJO GBMUTF46 HON	K	1

10 rows selected (18.933 seconds)

10. **Query to find the top 5 most occurred type of violation.**

```
CREATE TABLE tickets_description_2014
ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION
'/user/amahesh3/Group3/master2014/clean_nyc2014/tickets_description_2
014'
AS
select violation_description, count (violation_description) as
number_of_violation from clean_nyc2014 where violation_description !=
'' group by violation_description order by number_of_violation desc
limit 5;
```



violation_description	number_of_violation
38-Failure to Display Muni Rec	1230031
21-No Parking (street clean)	1086664
14-No Standing	753551
37-Expired Muni Meter	731685
FAILURE TO STOP AT RED LIGHT	433192

5 rows selected (17.325 seconds)

Step 8: Exporting data from HIVE

Use the below command to export the table from HDFS to Linux server,

```
hdfs dfs -get Group/master2014/clean_nyc2014/least_violated_location2014/000000_0
```

```
-bash-4.1$ hdfs dfs -get Group3/master2015/clean_nyc2015/least_violated_location2015/000000_0
-bash-4.1$
```

Open Windows power shell. Use the command showed below to export the data to the base machine in the form of .csv format

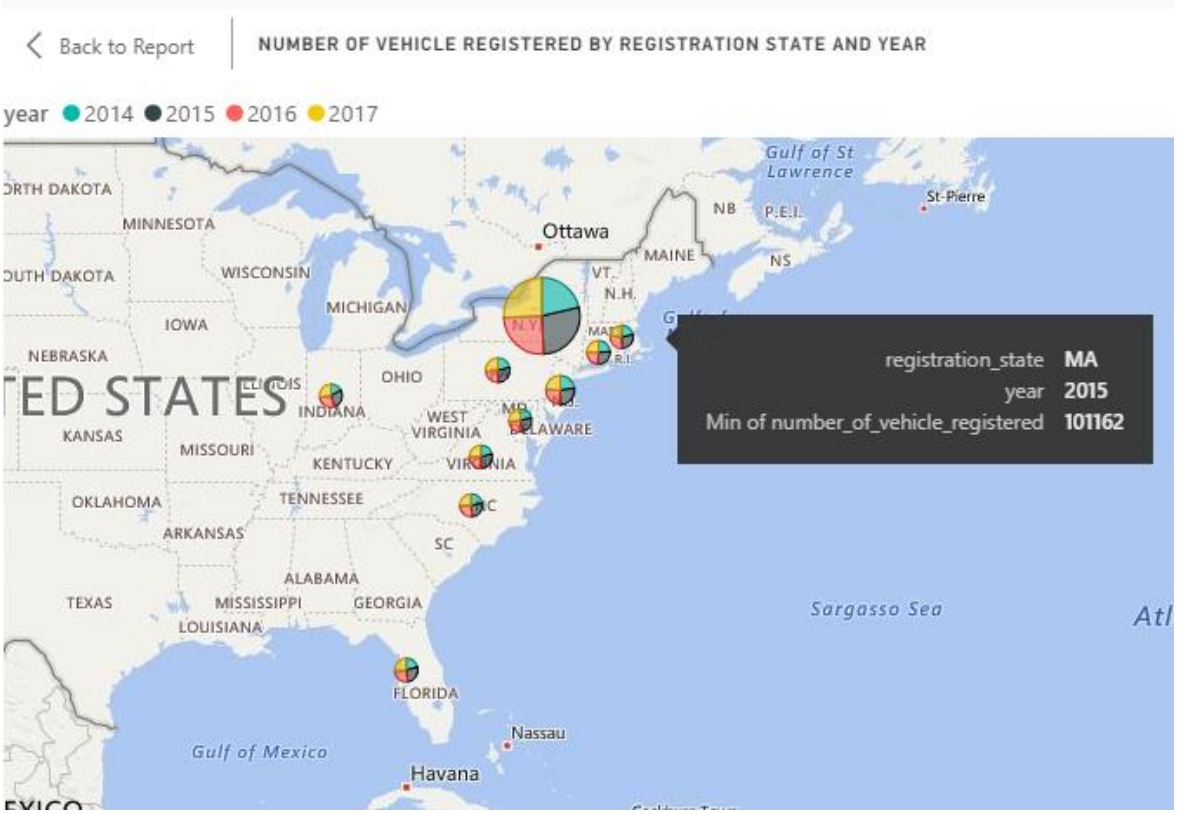
```
pscp -pw amahesh3 amahesh@129.150.129.67:/home/amahesh/000000_0 lease_violated_location2014.csv
```

```
PS E:\CSULA\CIS5200-01\Group 3\output> pscp -pw amahesh3 amahesh3@129.150.129.67:/home/amahesh3/000000_0 least_violated_
location2015.csv
least_violated_location20 | 0 kB |   0.2 kB/s | ETA: 00:00:00 | 100%
PS E:\CSULA\CIS5200-01\Group 3\output>
```

NOTE: Replace **2014** with other years. And to download csv files of other analyzed tables, first remove 000000_0 in hdfs using **hdfs dfs -rm 000000_0** command. Then get respective 000000_0 of required table and download it to local file using **pscp.**

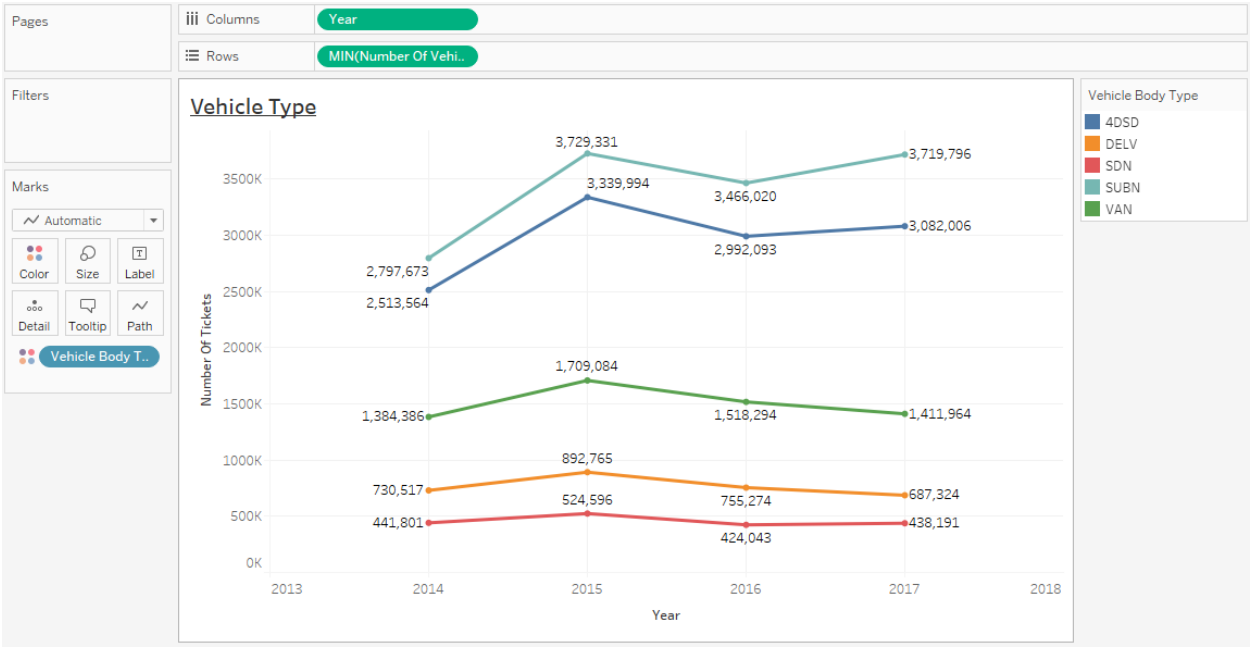
Step 9: Data Visualization using PowerBI, Tableau and 3D map

1. Open your PowerBI and upload the **vehicle_registered_state.csv** file.



Select registration_state as Location, year as Legend and size as number_of_violations. With this visualization we get to know how many violations are made by which registered state plate.

2. Open your Tableau to connect your server. You need to select **Text File** to open the file **violations_vehicle_type.csv**

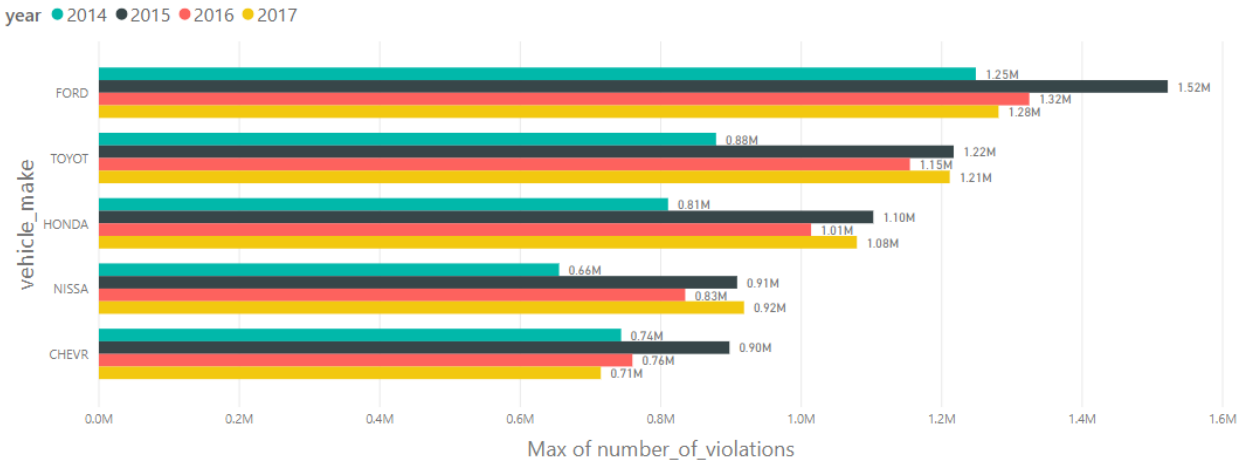


Select Year in Columns, number_of_vehicles in rows and vehicle_body_type in Legends. With this visualization we get to know which type of vehicle is getting more number of violation tickets.

3. Open your Tableau to connect your server. You need to select **Text File** to open the file **tickets_on_public_holidays.csv**

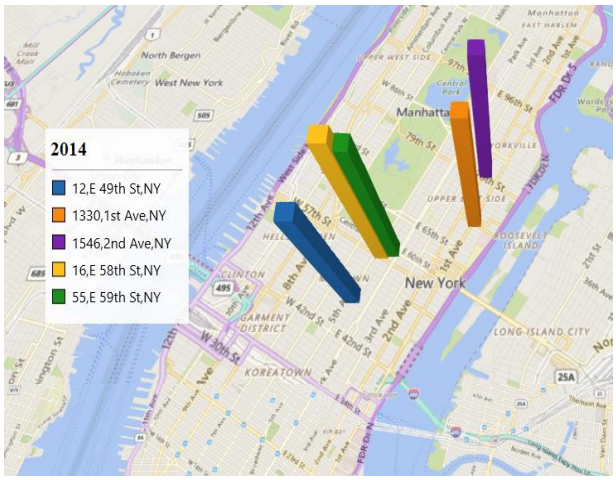
Select Year and Day in column, number of violations in rows and day as legends. The plot shows on which public holiday a greater number of violations were there.

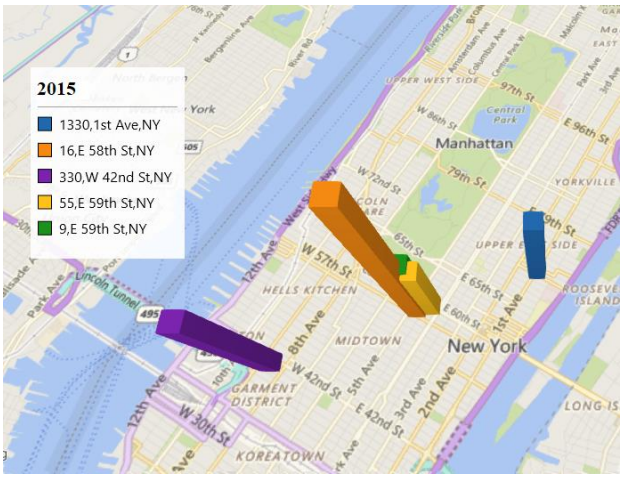
4. Open your PowerBI and upload the **violation_wrt_vehicle_make.csv** file.

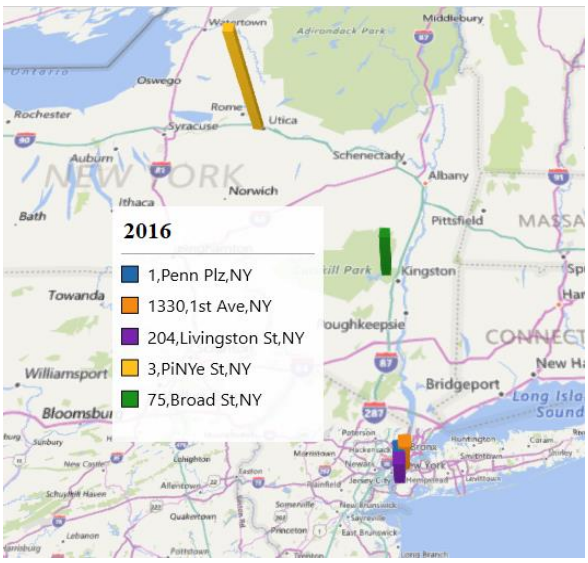


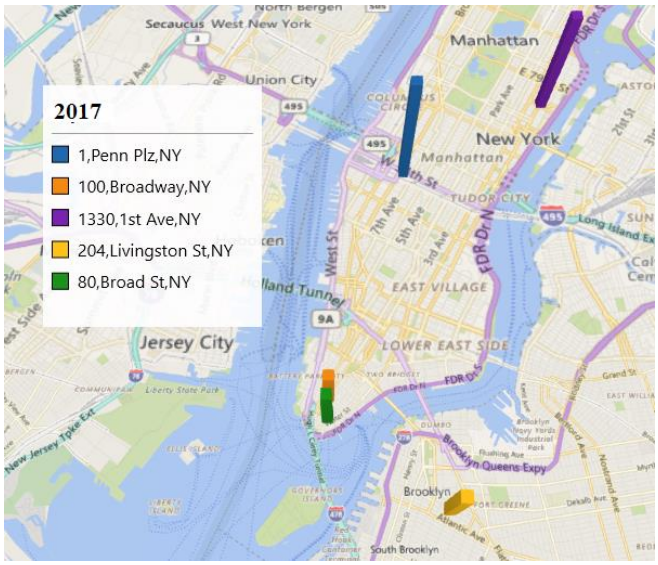
This plot depicts in all the 4 years which make of vehicle made a greater number of violations.

5. Convert **most_violated_location2014.csv, most_violated_location2015.csv, most_violated_location2016.csv and most_violated_location2017.csv** to xlsx format. Then go to insert -> 3D map. Select location as address and value as number of violations.



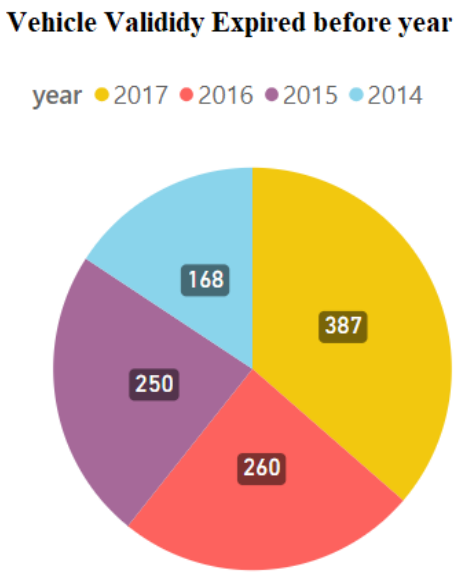






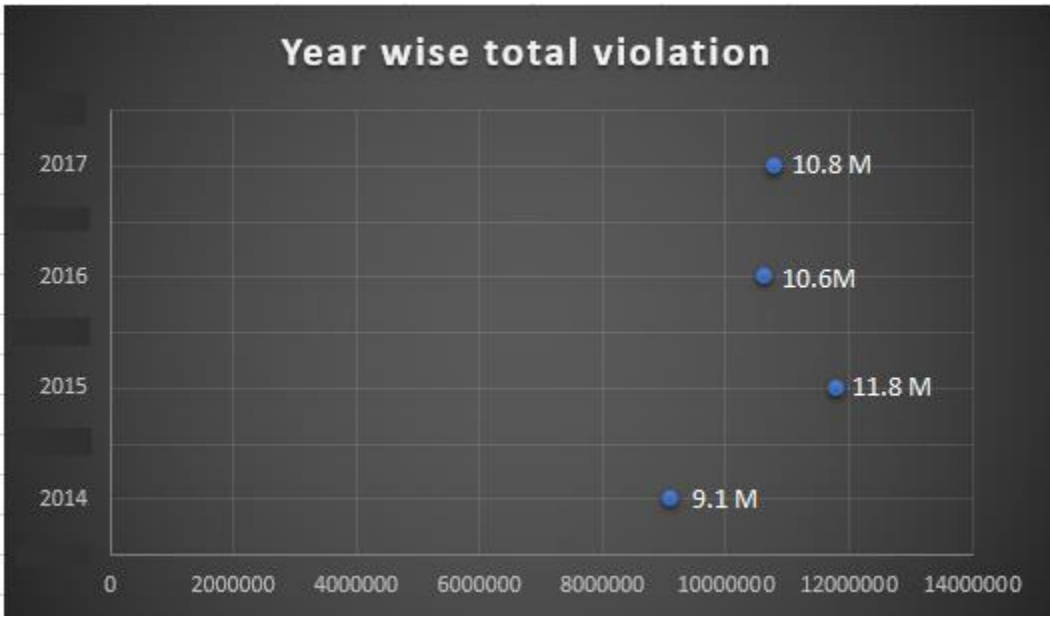
In this map we get to know at which location a greater number of violations has occurred.

6. Open your PowerBI and upload the **vehicle_count_expired.csv**.

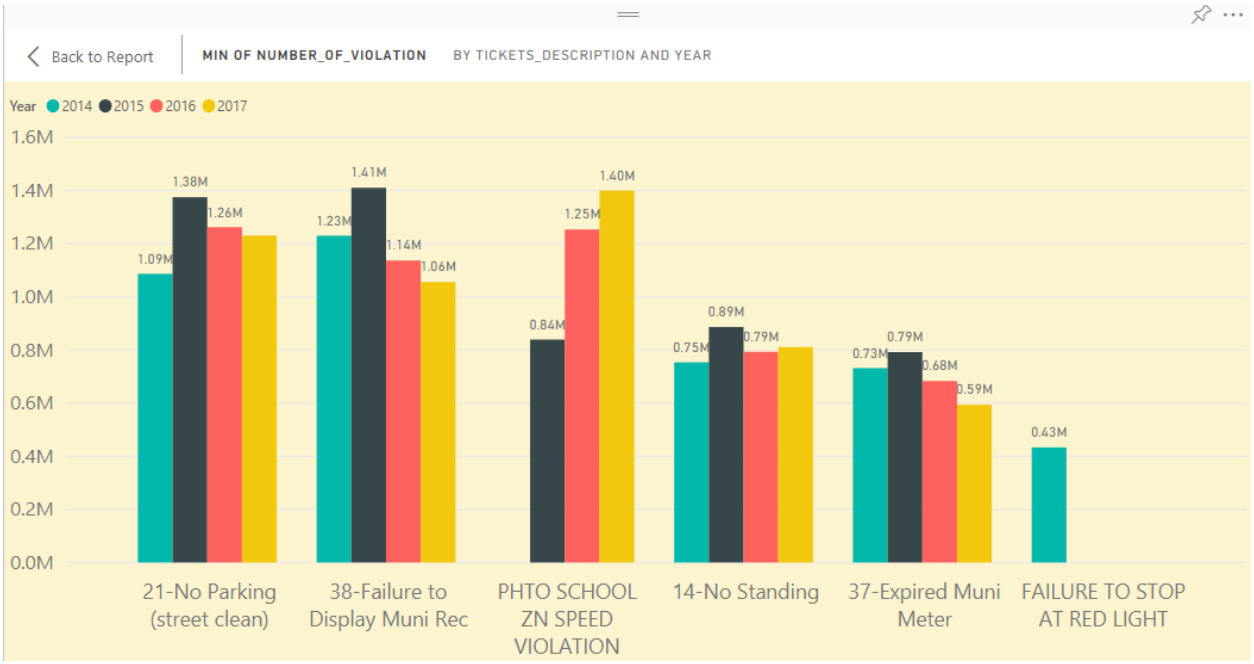


The pie chart shows in which year how many numbers of vehicles have expiration date before 2000.

7. This graph shows how many total violations have taken place.

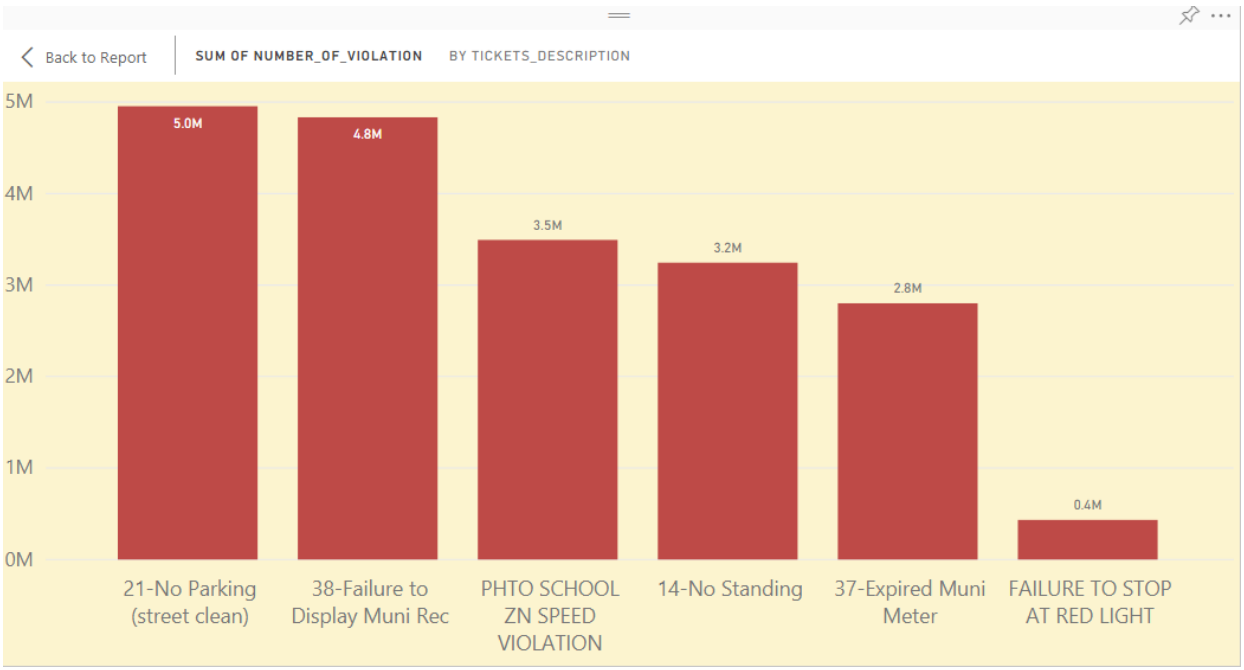


8. Upload **ticket_description.csv** to PowerBI and choose number_of_violations as values, violation_description as axis and year as legend.



This bar chart shows in which year which type of violation has occurred frequently.

9. With the same report, number_of_violations as values and violation_description as axis.



This chart shows which kind of violation has occurred the greatest number of times in all the four years.

References

https://github.com/amoghmahesh/hiveanalysisonnycparkingticket

https://www.kaggle.com/new-york-city/nyc-parking-tickets

http://www.nyc.gov/html/dof/html/pdf/faq/stars_codes.pdf

https://www.kaggle.com/donyoe/exploring-42-3m-nyc-parking-tickets/notebook

https://www1.nyc.gov/site/finance/vehicles/services-violation-codes.page