In [1]:
```
pwd
```

Out[1]:
```
'C:\\Users\\prade'
```

In [ ]:
```

```

(import the dataset using pandas) (INTRODUCTION-STEP 01)

In [2]:
```
import pandas as pd
```

In [4]:
```
USpro_dataset=pd.read_csv("USpro.csv")
```

call the dataset

In [5]:
```
USpro_dataset
```

Out[5]:

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Description	...	Roundabo
0	A-1	3	2016-02-08 00:37:08	2016-02-08 06:37:08	40.108910	-83.092860	40.112060	-83.031870	3.230	Between Sawmill Rd/Exit 20 and OH-315/Olentang...	...	Fa
1	A-2	2	2016-02-08 05:56:20	2016-02-08 11:56:20	39.865420	-84.062800	39.865010	-84.048730	0.747	At OH-4/OH-235/Exit 41 - Accident.	...	Fa
2	A-3	2	2016-02-08 06:15:39	2016-02-08 12:15:39	39.102660	-84.524680	39.102090	-84.523960	0.055	At I-71/US-50/Exit 1 - Accident.	...	Fa
3	A-4	2	2016-02-08 06:51:45	2016-02-08 12:51:45	41.062130	-81.537840	41.062170	-81.535470	0.123	At Dart Ave/Exit 21 - Accident.	...	Fa
4	A-5	3	2016-02-08 07:53:43	2016-02-08 13:53:43	39.172393	-84.492792	39.170476	-84.501798	0.500	At Mitchell Ave/Exit 6 - Accident.	...	Fa
...	...	...	...	...	...	...	...	...	...	...	...	
2845337	A-2845338	2	2019-08-23 18:03:25	2019-08-23 18:32:01	34.002480	-117.379360	33.998880	-117.370940	0.543	At Market St - Accident.	...	Fa
2845338	A-2845339	2	2019-08-23 19:11:30	2019-08-23 19:38:23	32.766960	-117.148060	32.765550	-117.153630	0.338	At Camino Del Rio/Mission Center Rd - Accident.	...	Fa
2845339	A-2845340	2	2019-08-23 19:00:21	2019-08-23 19:28:49	33.775450	-117.847790	33.777400	-117.857270	0.561	At Glassell St/Grand Ave - Accident. in the ri...	...	Fa
2845340	A-2845341	2	2019-08-23 19:00:21	2019-08-23 19:29:42	33.992460	-118.403020	33.983110	-118.395650	0.772	At CA-90/Marina Fwy/Jefferson Blvd - Accident.	...	Fa
2845341	A-2845342	2	2019-08-23 18:52:06	2019-08-23 19:21:31	34.133930	-117.230920	34.137360	-117.239340	0.537	At Highland Ave/Arden Ave - Accident.	...	Fa

2845342 rows × 47 columns

can call specific no of records by using using below given statement (check the header)

In [6]:
```
USpro_dataset.head(10)
```

Out[6]:

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Description	...	Roundabout	Station	S
0	A-1	3	2016-02-08 00:37:08	2016-02-08 06:37:08	40.108910	-83.092860	40.112060	-83.031870	3.230	Between Sawmill Rd/Exit 20 and OH-315/Olentang...	...	False	False	F
1	A-2	2	2016-02-08 05:56:20	2016-02-08 11:56:20	39.865420	-84.062800	39.865010	-84.048730	0.747	At OH-4/OH-235/Exit 41 - Accident.	...	False	False	F
2	A-3	2	2016-02-08 06:15:39	2016-02-08 12:15:39	39.102660	-84.524680	39.102090	-84.523960	0.055	At I-71/US-50/Exit 1 - Accident.	...	False	False	F
3	A-4	2	2016-02-08 06:51:45	2016-02-08 12:51:45	41.062130	-81.537840	41.062170	-81.535470	0.123	At Dart Ave/Exit 21 - Accident.	...	False	False	F
4	A-5	3	2016-02-08 07:53:43	2016-02-08 13:53:43	39.172393	-84.492792	39.170476	-84.501798	0.500	At Mitchell Ave/Exit 6 - Accident.	...	False	False	F
5	A-6	2	2016-02-08 08:16:57	2016-02-08 14:16:57	39.063240	-84.032430	39.067310	-84.058510	1.427	At Dela Palma Rd - Accident.	...	False	False	F
6	A-7	2	2016-02-08 08:15:41	2016-02-08 14:15:41	39.775650	-84.186030	39.772750	-84.188050	0.227	At OH-4/Exit 54 - Accident.	...	False	False	F
7	A-8	2	2016-02-08 11:51:46	2016-02-08 17:51:46	41.375310	-81.820170	41.367860	-81.821740	0.521	At Bagley Rd/Exit 235 - Accident.	...	False	False	F
8	A-9	2	2016-02-08 14:19:57	2016-02-08 20:19:57	40.702247	-84.075887	40.699110	-84.084293	0.491	At OH-65/Exit 122 - Accident.	...	False	False	F
9	A-10	2	2016-02-08 15:16:43	2016-02-08 21:16:43	40.109310	-82.968490	40.110780	-82.984000	0.826	At I-71/Exit 26 - Accident.	...	False	False	F

10 rows × 47 columns

to check 5 rows (check the tail)

In [7]:
```
USpro_dataset.tail(5)
```

Out[7]:

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Description	...	Roundabout	S
2845337	A-2845338	2	2019-08-23 18:03:25	2019-08-23 18:32:01	34.00248	-117.37936	33.99888	-117.37094	0.543	At Market St - Accident.	...	False	
2845338	A-2845339	2	2019-08-23 19:11:30	2019-08-23 19:38:23	32.76696	-117.14806	32.76555	-117.15363	0.338	At Camino Del Rio/Mission Center Rd - Accident.	...	False	
2845339	A-2845340	2	2019-08-23 19:00:21	2019-08-23 19:28:49	33.77545	-117.84779	33.77740	-117.85727	0.561	At Glassell St/Grand Ave - Accident. in the ri...	...	False	
2845340	A-2845341	2	2019-08-23 19:00:21	2019-08-23 19:29:42	33.99246	-118.40302	33.98311	-118.39565	0.772	At CA-90/Marina Fwy/Jefferson Blvd - Accident.	...	False	
2845341	A-2845342	2	2019-08-23 18:52:06	2019-08-23 19:21:31	34.13393	-117.23092	34.13736	-117.23934	0.537	At Highland Ave/Arden Ave - Accident.	...	False	

5 rows × 47 columns

US Accident Exploratory Data Analysis

TO DO - TALK ABOUT EDA; TO DO - Talk about the dataset(source,whit it contains,how it will be usefull);

1. Kaggle
2. information about accident
3. can be usefull to prevent accidents.
4. mention that this doesn't contain data about New York

STEP 02:

DATA PREPARATION AND CLEANING:

1. Load the file using pandas
2. Look at some information about the file
3. Fix any missing or incorrect datas or values.

LOAD THE FILE USING PANDAS

In [8]:
```
import pandas as pd
```

In [9]:
```
USpro_dataset=pd.read_csv("USpro.csv")
```

In [10]:
```
USpro_dataset
```

Out[10]:

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Description	...	Roundabo
0	A-1	3	2016-02-08 00:37:08	2016-02-08 06:37:08	40.108910	-83.092860	40.112060	-83.031870	3.230	Between Sawmill Rd/Exit 20 and OH-315/Olentang...	...	Fa
1	A-2	2	2016-02-08 05:56:20	2016-02-08 11:56:20	39.865420	-84.062800	39.865010	-84.048730	0.747	At OH-4/OH-235/Exit 41 - Accident.	...	Fa
2	A-3	2	2016-02-08 06:15:39	2016-02-08 12:15:39	39.102660	-84.524680	39.102090	-84.523960	0.055	At I-71/US-50/Exit 1 - Accident.	...	Fa
3	A-4	2	2016-02-08 06:51:45	2016-02-08 12:51:45	41.062130	-81.537840	41.062170	-81.535470	0.123	At Dart Ave/Exit 21 - Accident.	...	Fa
4	A-5	3	2016-02-08 07:53:43	2016-02-08 13:53:43	39.172393	-84.492792	39.170476	-84.501798	0.500	At Mitchell Ave/Exit 6 - Accident.	...	Fa
...	...	...	...	...	...	...	...	...	...	...	...	
2845337	A-2845338	2	2019-08-23 18:03:25	2019-08-23 18:32:01	34.002480	-117.379360	33.998880	-117.370940	0.543	At Market St - Accident.	...	Fa
2845338	A-2845339	2	2019-08-23 19:11:30	2019-08-23 19:38:23	32.766960	-117.148060	32.765550	-117.153630	0.338	At Camino Del Rio/Mission Center Rd - Accident.	...	Fa
2845339	A-2845340	2	2019-08-23 19:00:21	2019-08-23 19:28:49	33.775450	-117.847790	33.777400	-117.857270	0.561	At Glassell St/Grand Ave - Accident. in the ri...	...	Fa
2845340	A-2845341	2	2019-08-23 19:00:21	2019-08-23 19:29:42	33.992460	-118.403020	33.983110	-118.395650	0.772	At CA-90/Marina Fwy/Jefferson Blvd - Accident.	...	Fa
2845341	A-2845342	2	2019-08-23 18:52:06	2019-08-23 19:21:31	34.133930	-117.230920	34.137360	-117.239340	0.537	At Highland Ave/Arden Ave - Accident.	...	Fa

2845342 rows × 47 columns

LOOK AT SOME INFORMATION ABOUT THE FILE

In [13]:
```
USpro_dataset.columns
```

Out[13]:
```
Index(['ID', 'Severity', 'Start_Time', 'End_Time', 'Start_Lat', 'Start_Lng',
       'End_Lat', 'End_Lng', 'Distance(mi)', 'Description', 'Number', 'Street',
       'Side', 'City', 'County', 'State', 'Zipcode', 'Country', 'Timezone',
       'Airport_Code', 'Weather_Timestamp', 'Temperature(F)', 'Wind_Chill(F)',
       'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Direction',
       'Wind_Speed(mph)', 'Precipitation(in)', 'Weather_Condition', 'Amenity',
       'Bump', 'Crossing', 'Give_Way', 'Junction', 'No_Exit', 'Railway',
       'Roundabout', 'Station', 'Stop', 'Traffic_Calming', 'Traffic_Signal',
       'Turning_Loop', 'Sunrise_Sunset', 'Civil_Twilight', 'Nautical_Twilight',
       'Astronomical_Twilight'],
      dtype='object')
```

In [14]:
```
USpro_dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2845342 entries, 0 to 2845341
Data columns (total 47 columns):
 #   Column                 Dtype
---  ------                 -----
 0   ID                     object
 1   Severity               int64
 2   Start_Time             object
 3   End_Time               object
 4   Start_Lat              float64
 5   Start_Lng              float64
 6   End_Lat                float64
 7   End_Lng                float64
 8   Distance(mi)           float64
 9   Description            object
 10  Number                 float64
 11  Street                 object
 12  Side                   object
 13  City                   object
 14  County                 object
 15  State                  object
 16  Zipcode                object
 17  Country                object
 18  Timezone               object
 19  Airport_Code           object
 20  Weather_Timestamp      object
 21  Temperature(F)         float64
 22  Wind_Chill(F)          float64
 23  Humidity(%)            float64
 24  Pressure(in)           float64
 25  Visibility(mi)         float64
 26  Wind_Direction         object
 27  Wind_Speed(mph)        float64
 28  Precipitation(in)      float64
 29  Weather_Condition      object
 30  Amenity                bool
 31  Bump                   bool
 32  Crossing               bool
 33  Give_Way               bool
 34  Junction               bool
 35  No_Exit                bool
 36  Railway                bool
 37  Roundabout             bool
 38  Station                bool
 39  Stop                   bool
 40  Traffic_Calming        bool
 41  Traffic_Signal         bool
 42  Turning_Loop           bool
 43  Sunrise_Sunset         object
 44  Civil_Twilight         object
 45  Nautical_Twilight      object
 46  Astronomical_Twilight  object
dtypes: bool(13), float64(13), int64(1), object(20)
memory usage: 773.4+ MB
```

calculate summary statistics

In [15]:
```
USpro_dataset.describe()
```

Out[15]:

	Severity	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Number	Temperature(F)	Wind_Chill(F)
count	2.845342e+06	2.845342e+06	2.845342e+06	2.845342e+06	2.845342e+06	2.845342e+06	1.101431e+06	2.776068e+06	2.375699e+06
mean	2.137572e+00	3.624520e+01	-9.711463e+01	3.624532e+01	-9.711439e+01	7.026779e-01	8.089408e+03	6.179356e+01	5.965823e+01
std	4.787216e-01	5.363797e+00	1.831782e+01	5.363873e+00	1.831763e+01	1.560361e+00	1.836009e+04	1.862263e+01	2.116097e+01
min	1.000000e+00	2.456603e+01	-1.245481e+02	2.456601e+01	-1.245457e+02	0.000000e+00	0.000000e+00	-8.900000e+01	-8.900000e+01
25%	2.000000e+00	3.344517e+01	-1.180331e+02	3.344628e+01	-1.180333e+02	5.200000e-02	1.270000e+03	5.000000e+01	4.600000e+01
50%	2.000000e+00	3.609861e+01	-9.241808e+01	3.609799e+01	-9.241772e+01	2.440000e-01	4.007000e+03	6.400000e+01	6.300000e+01
75%	2.000000e+00	4.016024e+01	-8.037243e+01	4.016105e+01	-8.037338e+01	7.640000e-01	9.567000e+03	7.600000e+01	7.600000e+01
max	4.000000e+00	4.900058e+01	-6.711317e+01	4.907500e+01	-6.710924e+01	1.551860e+02	9.999997e+06	1.960000e+02	1.960000e+02

KEEP ANALYSING AND RAISE THE QUE HERE:

ask and answer questions

1. are there more accidents in warmer or colder areas?
2. which 5 states have the highest number of accidents ? how about per capital?
3. does new york show up in the data ? if yes, why is the count lower if this the most populated city.

how pandas count number of numerical columns

In [19]:
```
numerics=['int16','int32','int64','float16','float32','float64']
numeric_USpro_dataset=USpro_dataset.select_dtypes(include=numerics)
len(numeric_USpro_dataset.columns)
```

Out[19]: 14

hence,we have 14 numeric columns rest of the data will probably dates or categorical data etc..

followed , finding percentage of missing values in pandas

filename.isna()

each values get replaced by true or false

1. it gets replaced by true , if this value is any null or empty on missing.
2. it gets replaced by false , if this value does exist and it doesn't empty.

In [21]:
```
USpro_dataset.isna()
```

Out[21]:

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Description	...	Roundabout	Station
0	False	False	False	False	False	False	False	False	False	False	...	False	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False
3	False	False	False	False	False	False	False	False	False	False	...	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2845337	False	False	False	False	False	False	False	False	False	False	...	False	False
2845338	False	False	False	False	False	False	False	False	False	False	...	False	False
2845339	False	False	False	False	False	False	False	False	False	False	...	False	False
2845340	False	False	False	False	False	False	False	False	False	False	...	False	False
2845341	False	False	False	False	False	False	False	False	False	False	...	False	False

2845342 rows × 47 columns

filename.isna().sum()

gives per column the count of missing values

In [22]:
```
USpro_dataset.isna().sum()
```

Out[22]:
```
ID                          0
Severity                    0
Start_Time                  0
End_Time                    0
Start_Lat                   0
Start_Lng                   0
End_Lat                     0
End_Lng                     0
Distance(mi)                0
Description                 0
Number                1743911
Street                      2
Side                        0
City                      137
County                      0
State                       0
Zipcode                  1319
Country                     0
Timezone                 3659
Airport_Code             9549
Weather_Timestamp       50736
Temperature(F)          69274
Wind_Chill(F)          469643
Humidity(%)             73092
Pressure(in)            59200
Visibility(mi)          70546
Wind_Direction          73775
Wind_Speed(mph)        157944
Precipitation(in)      549458
Weather_Condition       70636
Amenity                     0
Bump                        0
Crossing                    0
Give_Way                    0
Junction                    0
No_Exit                     0
Railway                     0
Roundabout                  0
Station                     0
Stop                        0
Traffic_Calming             0
Traffic_Signal              0
Turning_Loop                0
Sunrise_Sunset           2867
Civil_Twilight           2867
Nautical_Twilight        2867
Astronomical_Twilight    2867
dtype: int64
```

sorting no of missing values in decending order

In [24]:
```
USpro_dataset.isna().sum().sort_values(ascending=False)
```

Out[24]:

```
Number                   1743911
Precipitation(in)         549458
Wind_Chill(F)             469643
Wind_Speed(mph)           157944
Wind_Direction             73775
Humidity(%)                73092
Weather_Condition          70636
Visibility(mi)             70546
Temperature(F)             69274
Pressure(in)               59200
Weather_Timestamp          50736
Airport_Code                9549
Timezone                    3659
Nautical_Twilight           2867
Civil_Twilight              2867
Sunrise_Sunset              2867
Astronomical_Twilight       2867
Zipcode                     1319
City                         137
Street                         2
Country                        0
Junction                       0
Start_Time                     0
End_Time                       0
Start_Lat                      0
Turning_Loop                   0
Traffic_Signal                 0
Traffic_Calming                0
Stop                           0
Station                        0
Roundabout                     0
Railway                        0
No_Exit                        0
Crossing                       0
Give_Way                       0
Bump                           0
Amenity                        0
Start_Lng                      0
End_Lat                        0
End_Lng                        0
Distance(mi)                   0
Description                    0
Severity                       0
Side                           0
County                         0
State                          0
ID                             0
dtype: int64
```

missing percentage:

In [25]:

```
missing_percentage = USpro_dataset.isna().sum().sort_values(ascending=False) / len(USpro_dataset)
missing_percentage
```

Out[25]:
```
Number                  6.129003e-01
Precipitation(in)       1.931079e-01
Wind_Chill(F)           1.650568e-01
Wind_Speed(mph)         5.550967e-02
Wind_Direction          2.592834e-02
Humidity(%)             2.568830e-02
Weather_Condition       2.482514e-02
Visibility(mi)          2.479350e-02
Temperature(F)          2.434646e-02
Pressure(in)            2.080593e-02
Weather_Timestamp       1.783125e-02
Airport_Code            3.356011e-03
Timezone                1.285961e-03
Nautical_Twilight       1.007612e-03
Civil_Twilight          1.007612e-03
Sunrise_Sunset          1.007612e-03
Astronomical_Twilight   1.007612e-03
Zipcode                 4.635647e-04
City                    4.814887e-05
Street                  7.029032e-07
Country                 0.000000e+00
Junction                0.000000e+00
Start_Time              0.000000e+00
End_Time                0.000000e+00
Start_Lat               0.000000e+00
Turning_Loop            0.000000e+00
Traffic_Signal          0.000000e+00
Traffic_Calming         0.000000e+00
Stop                    0.000000e+00
Station                 0.000000e+00
Roundabout              0.000000e+00
Railway                 0.000000e+00
No_Exit                 0.000000e+00
Crossing                0.000000e+00
Give_Way                0.000000e+00
Bump                    0.000000e+00
Amenity                 0.000000e+00
Start_Lng               0.000000e+00
End_Lat                 0.000000e+00
End_Lng                 0.000000e+00
Distance(mi)            0.000000e+00
Description             0.000000e+00
Severity                0.000000e+00
Side                    0.000000e+00
County                  0.000000e+00
State                   0.000000e+00
ID                      0.000000e+00
dtype: float64
```

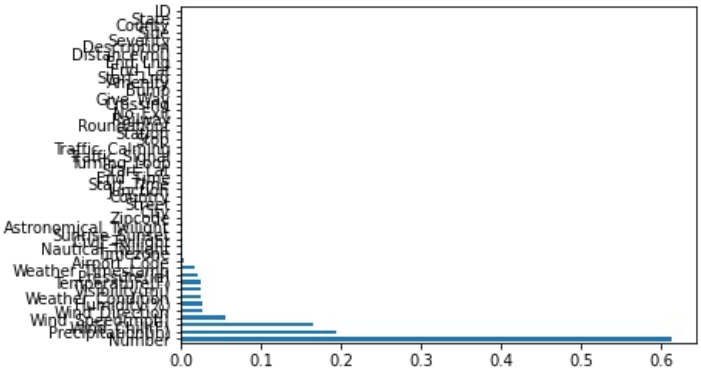
In [26]:
```
type(missing_percentage)
```

Out[26]:
```
pandas.core.series.Series
```

visualizing the percentage of missing values in graph:

In [27]:
```
missing_percentage.plot(kind='barh')
```

Out[27]:
```
<AxesSubplot:>
```



filter zeroes from missing percentage

In [28]:
```
missing_percentage !=0
```

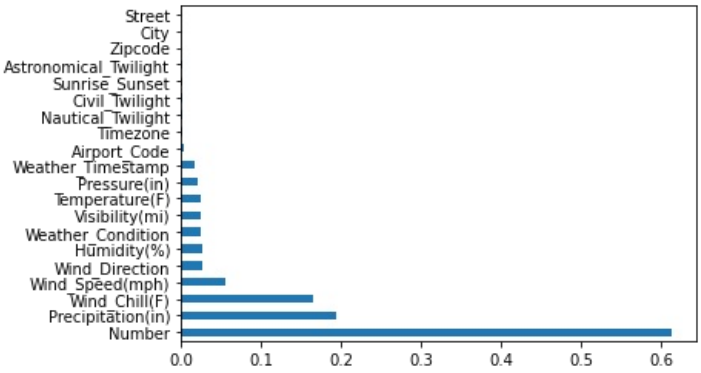
Out[28]:
```
Number                   True
Precipitation(in)        True
Wind_Chill(F)            True
Wind_Speed(mph)          True
Wind_Direction           True
Humidity(%)              True
Weather_Condition        True
Visibility(mi)           True
Temperature(F)           True
Pressure(in)             True
Weather_Timestamp        True
Airport_Code             True
Timezone                 True
Nautical_Twilight        True
Civil_Twilight           True
Sunrise_Sunset           True
Astronomical_Twilight    True
Zipcode                  True
City                     True
Street                   True
Country                 False
Junction                False
Start_Time              False
End_Time                False
Start_Lat               False
Turning_Loop            False
Traffic_Signal          False
Traffic_Calming         False
Stop                    False
Station                 False
Roundabout              False
Railway                 False
No_Exit                 False
Crossing                False
Give_Way                False
Bump                    False
Amenity                 False
Start_Lng               False
End_Lat                 False
End_Lng                 False
Distance(mi)            False
Description             False
Severity                False
Side                    False
County                  False
State                   False
ID                      False
dtype: bool
```

In [31]:
```
missing_percentage [missing_percentage != 0]
```

Out[31]:
```
Number                   6.129003e-01
Precipitation(in)        1.931079e-01
Wind_Chill(F)            1.650568e-01
Wind_Speed(mph)          5.550967e-02
Wind_Direction           2.592834e-02
Humidity(%)              2.568830e-02
Weather_Condition        2.482514e-02
Visibility(mi)           2.479350e-02
Temperature(F)           2.434646e-02
Pressure(in)             2.080593e-02
Weather_Timestamp        1.783125e-02
Airport_Code             3.356011e-03
Timezone                 1.285961e-03
Nautical_Twilight        1.007612e-03
Civil_Twilight           1.007612e-03
Sunrise_Sunset           1.007612e-03
Astronomical_Twilight    1.007612e-03
Zipcode                  4.635647e-04
City                     4.814887e-05
Street                   7.029032e-07
dtype: float64
```

In [35]:
```
missing_percentage [missing_percentage != 0].plot(kind='barh')
```

Out[35]:
```
<AxesSubplot:>
```



remove the columns that we don't want to use.

STEP 3:

EXPLORATORY ANALYSIS AND VISUALIZATION

TO DO - Pick 4 or 5 interesting columns & for each of those columns we will just plot some graphs and see what they look like.

1. city
2. start_time
3. start_lat,start_lng
4. temparature
5. whether_condition

In [36]:
```
USpro_dataset.columns
```

Out[36]:
```
Index(['ID', 'Severity', 'Start_Time', 'End_Time', 'Start_Lat', 'Start_Lng',
       'End_Lat', 'End_Lng', 'Distance(mi)', 'Description', 'Number', 'Street',
       'Side', 'City', 'County', 'State', 'Zipcode', 'Country', 'Timezone',
       'Airport_Code', 'Weather_Timestamp', 'Temperature(F)', 'Wind_Chill(F)',
       'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Direction',
       'Wind_Speed(mph)', 'Precipitation(in)', 'Weather_Condition', 'Amenity',
       'Bump', 'Crossing', 'Give_Way', 'Junction', 'No_Exit', 'Railway',
       'Roundabout', 'Station', 'Stop', 'Traffic_Calming', 'Traffic_Signal',
       'Turning_Loop', 'Sunrise_Sunset', 'Civil_Twilight', 'Nautical_Twilight',
       'Astronomical_Twilight'],
      dtype='object')
```

In [38]:
```
USpro_dataset.City
```

Out[38]:
```
0               Dublin
1               Dayton
2           Cincinnati
3                Akron
4           Cincinnati
              ...
2845337      Riverside
2845338      San Diego
2845339         Orange
2845340    Culver City
2845341       Highland
Name: City, Length: 2845342, dtype: object
```

In [42]:
```
cities = USpro_dataset.City.unique()
len(cities)
```

Out[42]:
```
11682
```

look at the cities where major accident occur

1. count no of accients occur in each cities , named as (cities_by_accident)

In [47]:
```
cities_by_accident = USpro_dataset.City.value_counts()
cities_by_accident
```

Out[47]:
```
Miami                              106966
Los Angeles                         68956
Orlando                             54691
Dallas                              41979
Houston                             39448
                                    ...
Ridgedale                               1
Sekiu                                   1
Wooldridge                              1
Bullock                                 1
American Fork-Pleasant Grove            1
Name: City, Length: 11681, dtype: int64
```

look at top twenty

In [49]:
```
cities_by_accident[:20]
```

Out[49]:
```
Miami           106966
Los Angeles      68956
Orlando          54691
Dallas           41979
Houston          39448
Charlotte        33152
Sacramento       32559
San Diego        26627
Raleigh          22840
Minneapolis      22768
Portland         20944
Nashville        20267
Austin           18301
Baton Rouge      18182
Phoenix          17143
Saint Paul       16869
New Orleans      16251
Atlanta          15622
Jacksonville     14967
Richmond         14349
Name: City, dtype: int64
```

In [53]:
```
'New York' in USpro_dataset.City
```

Out[53]:
```
False
```

In [55]:
```
'New York' in USpro_dataset.State
```

Out[55]:
```
False
```

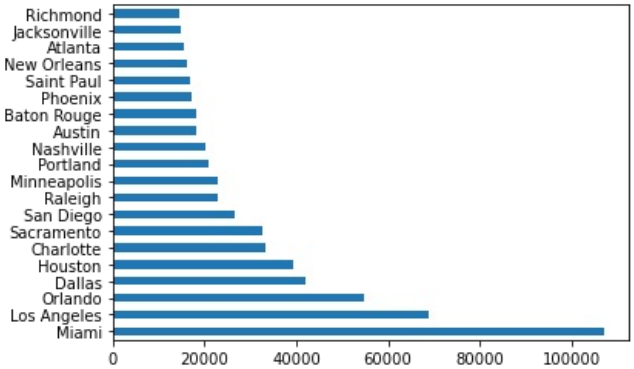
KEEP ANALYSING AND RAISE THE QUE HERE:

ask and answer questions

1. are there more accidents in warmer or colder areas?
2. which 5 states have the highest number of accidents ? how about per capital?
3. does new york show up in the data ? if yes, why is the count lower if this the most populated city.

In [58]:
```
cities_by_accident[:20].plot(kind='barh')
```

Out[58]:
```
<AxesSubplot:>
```



KEEP ANALYSING AND RAISE THE QUE HERE:

ask and answer questions

1. are there more accidents in warmer or colder areas?
2. which 5 states have the highest number of accidents ? how about per capital?
3. does new york show up in the data ? if yes, why is the count lower if this the most populated city.
4. Among the top 100 cities in number of accidents, which states do they belong to most frequently.

do a lot of cities have a small number of accidents or do lot of cities have high number of accidents , what does that distribution look like?

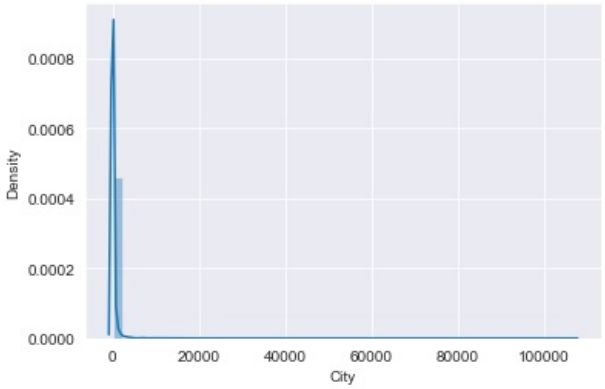
And the way to do that is to use a histogram plot.(using darkgrid theme here)

In [60]:
```
import seaborn as sns
sns.set_style("darkgrid")
```

In [63]:
```
sns.distplot(cities_by_accident)
```

```
C:\Users\prade\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprec
ated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure
-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[63]:
```
<AxesSubplot:xlabel='City', ylabel='Density'>
```



In [64]:
```
high_accident_cities = cities_by_accident[cities_by_accident >= 1000]
low_accident_cities = cities_by_accident[cities_by_accident < 1000 ]
```

In [65]:
```
len(high_accident_cities)
```

Out[65]:
```
496
```

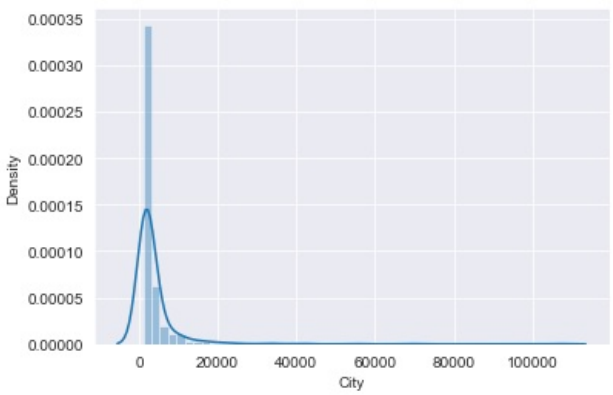
In [68]:
```
len(high_accident_cities) / len(cities)
```

Out[68]:
```
0.04245848313644924
```

In [72]:
```
sns.distplot(high_accident_cities)
```

```
C:\Users\prade\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprec
ated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure
-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

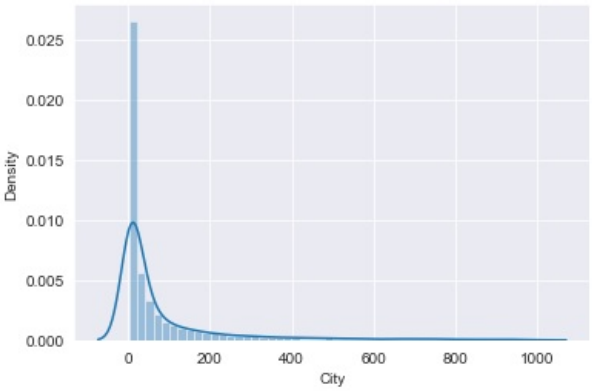
Out[72]:
```
<AxesSubplot:xlabel='City', ylabel='Density'>
```



In [70]:
```
sns.distplot(low_accident_cities)
```

```
C:\Users\prade\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprec
ated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure
-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

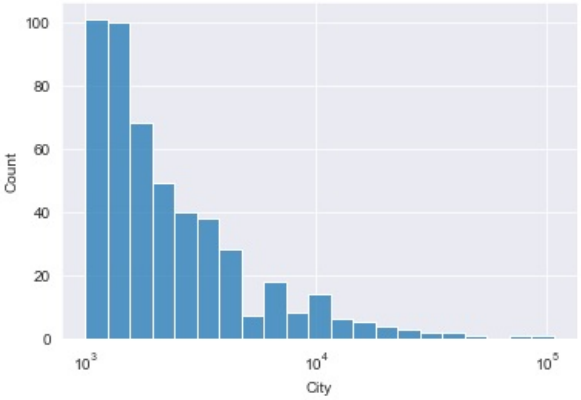
Out[70]:
```
<AxesSubplot:xlabel='City', ylabel='Density'>
```



as distplot doesn't show much difference log scale has been included using hist

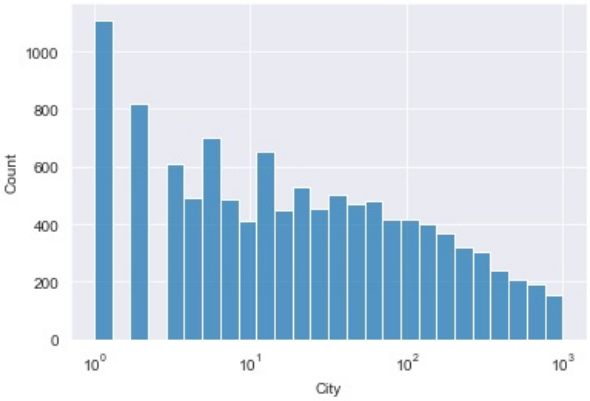
In [73]:
```
sns.histplot(high_accident_cities,log_scale=True)
```

Out[73]:
```
<AxesSubplot:xlabel='City', ylabel='Count'>
```



In [74]:
```
sns.histplot(low_accident_cities,log_scale=True)
```

Out[74]:
```
<AxesSubplot:xlabel='City', ylabel='Count'>
```



In [84]:
```
cities_by_accident[cities_by_accident== 1]
```

Out[84]:
```
Carney                          1
Waverly Hall                    1
Center Sandwich                 1
Glen Flora                      1
Sulphur Springs                 1
                               ..
Ridgedale                       1
Sekiu                           1
Wooldridge                      1
Bullock                         1
American Fork-Pleasant Grove    1
Name: City, Length: 1110, dtype: int64
```

START TIME: (TOPIC 2)

In [86]:
```
USpre_dataset_columns
```

In [86]:
```
USpro_dataset.columns
```

Out[86]:
```
Index(['ID', 'Severity', 'Start_Time', 'End_Time', 'Start_Lat', 'Start_Lng',
       'End_Lat', 'End_Lng', 'Distance(mi)', 'Description', 'Number', 'Street',
       'Side', 'City', 'County', 'State', 'Zipcode', 'Country', 'Timezone',
       'Airport_Code', 'Weather_Timestamp', 'Temperature(F)', 'Wind_Chill(F)',
       'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Direction',
       'Wind_Speed(mph)', 'Precipitation(in)', 'Weather_Condition', 'Amenity',
       'Bump', 'Crossing', 'Give_Way', 'Junction', 'No_Exit', 'Railway',
       'Roundabout', 'Station', 'Stop', 'Traffic_Calming', 'Traffic_Signal',
       'Turning_Loop', 'Sunrise_Sunset', 'Civil_Twilight', 'Nautical_Twilight',
       'Astronomical_Twilight'],
      dtype='object')
```

In [87]:
```
USpro_dataset.Start_Time
```

Out[87]:
```
0          2016-02-08 00:37:08
1          2016-02-08 05:56:20
2          2016-02-08 06:15:39
3          2016-02-08 06:51:45
4          2016-02-08 07:53:43
                  ...
2845337    2019-08-23 18:03:25
2845338    2019-08-23 19:11:30
2845339    2019-08-23 19:00:21
2845340    2019-08-23 19:00:21
2845341    2019-08-23 18:52:06
Name: Start_Time, Length: 2845342, dtype: object
```

In [88]:
```
USpro_dataset.Start_Time[0]
```

Out[88]:
```
'2016-02-08 00:37:08'
```

currently it is in string so convert it in date.

In [90]:
```
USpro_dataset.Start_Time = pd.to_datetime(USpro_dataset.Start_Time)
```

In [91]:
```
USpro_dataset.Start_Time[0]
```

Out[91]:
```
Timestamp('2016-02-08 00:37:08')
```

KEEP ANALYSING AND RAISE THE QUE HERE:

ask and answer questions

1. are there more accidents in warmer or colder areas?
2. which 5 states have the highest number of accidents ? how about per capital?
3. does new york show up in the data ? if yes, why is the count lower if this the most populated city.
4. Among the top 100 cities in number of accidents, which states do they belong to most frequently.
5. what time of the day are accidents most frequent in?
6. which days of the week have the most accidents?
7. which month have the most accident?
8. what is the trend of accidents year over year (decreasing\increasing)?

In [96]:
```
USpro_dataset.Start_Time[4].hour
```

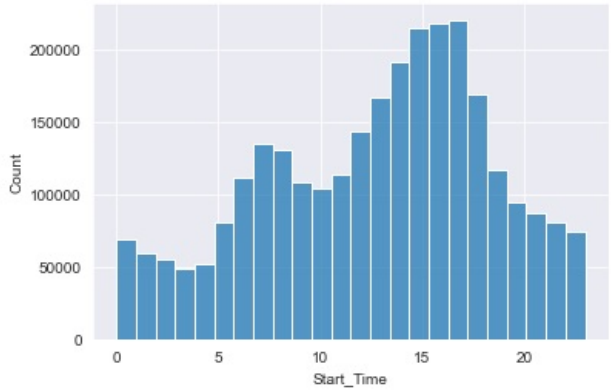
Out[96]:
```
7
```

In [98]:
```
USpro_dataset.Start_Time.dt.hour
```

Out[98]:
```
0          0
1          5
2          6
3          6
4          7
          ..
2845337    18
2845338    19
2845339    19
2845340    19
2845341    18
Name: Start_Time, Length: 2845342, dtype: int64
```

In [101…
```
sns.histplot(USpro_dataset.Start_Time.dt.hour,bins=24)
```

Out[101]:
```
<AxesSubplot:xlabel='Start_Time', ylabel='Count'>
```

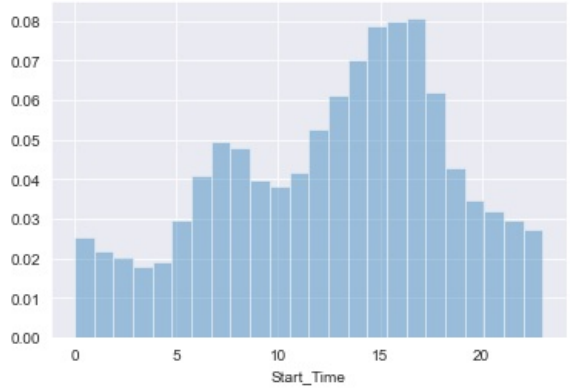


change y axis in percentage

In [102…
```
sns.distplot(USpro_dataset.Start_Time.dt.hour,bins=24,kde = False, norm_hist= True)
```

```
C:\Users\prade\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprec
ated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure
-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[102]:
```
<AxesSubplot:xlabel='Start_Time'>
```



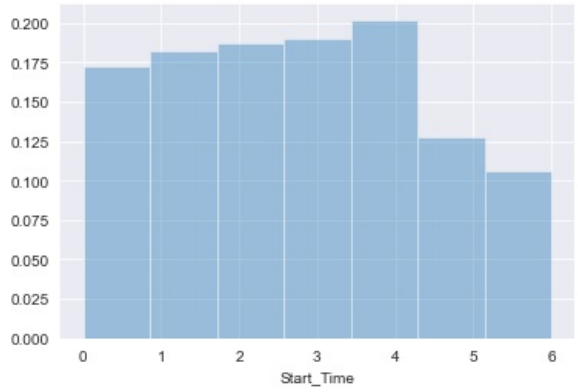
High percentage of accident occurs between 12pm-6pm(probably due to eve time and people in hurry to get to home and parties)

next highest percentage is from 6am to 9am (people in a hurry to get to work)

day of week

In [107…
```
sns.distplot(USpro_dataset.Start_Time.dt.dayofweek,bins=7,kde = False, norm_hist= True)
```

Out[107]:
```
<AxesSubplot:xlabel='Start_Time'>
```



on weekends the number of accidents are lower compared to week days.

KEEP ANALYSING AND RAISE THE QUE HERE:

ask and answer questions

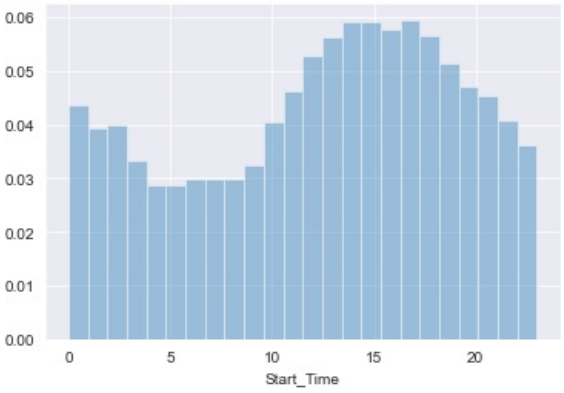
1. are there more accidents in warmer or colder areas?
2. which 5 states have the highest number of accidents ? how about per capital?
3. does new york show up in the data ? if yes, why is the count lower if this the most populated city.
4. Among the top 100 cities in number of accidents, which states do they belong to most frequently.
5. what time of the day are accidents most frequent in? = ANSWERED
6. which days of the week have the most accidents?
7. which month have the most accident?
8. what is the trend of accidents year over year (decreasing\increasing)?
9. Is the distribution of accident by hour the same on weekends as on weekdays ?

In [109…
```
sunday_Start_Time = USpro_dataset.Start_Time[USpro_dataset.Start_Time.dt.dayofweek == 6]
```

In [112…
```
sns.distplot(sunday_Start_Time.dt.hour,bins=24,kde = False, norm_hist= True)
```

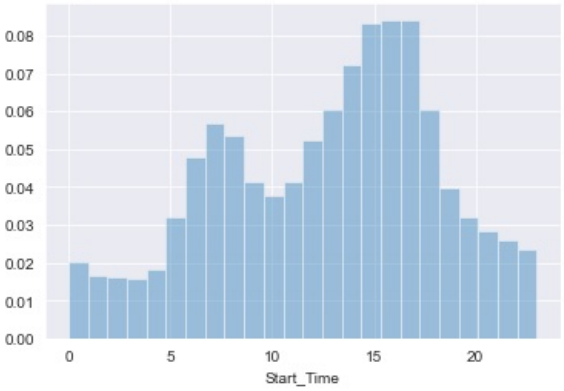
```
C:\Users\prade\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprec
ated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure
-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[112]:
```
<AxesSubplot:xlabel='Start_Time'>
```



In [113…
```
monday_Start_Time = USpro_dataset.Start_Time[USpro_dataset.Start_Time.dt.dayofweek == 0]
sns.distplot(monday_Start_Time.dt.hour,bins=24,kde = False, norm_hist= True)
```

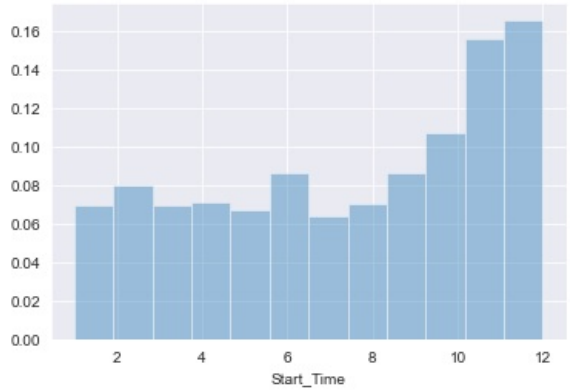
Out[113]:
```
<AxesSubplot:xlabel='Start_Time'>
```



on sundays the peak occurs between 10amto 10pm unlike weekdays

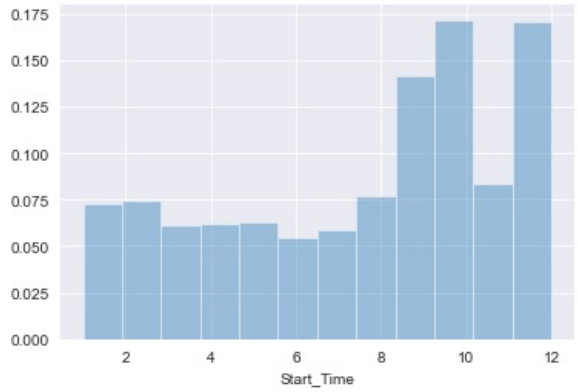
In [117…
```
sns.distplot(monday_Start_Time.dt.month,bins=12,kde = False, norm_hist= True)
```

Out[117]:
```
<AxesSubplot:xlabel='Start_Time'>
```



In [142…
```
USpro_dataset_2019 = USpro_dataset[USpro_dataset.Start_Time.dt.year == 2019]
sns.distplot(USpro_dataset_2019.Start_Time.dt.month,bins=12,kde = False, norm_hist= True)
```

Out[142]:
```
<AxesSubplot:xlabel='Start_Time'>
```



much data is missing for 2016 and 2020 ,may be even 2017

start_latitude and start_longitude: TOPIC 3

In [154…
```
USpro_dataset.Start_Lat
```

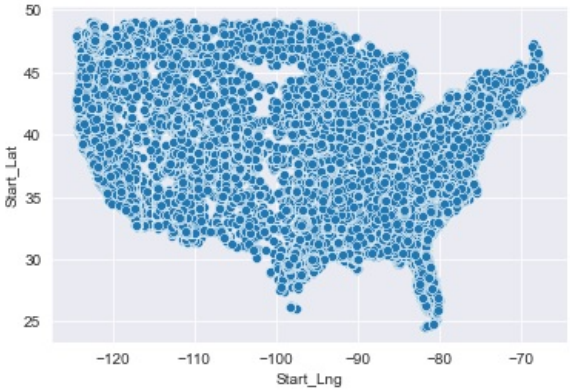
Out[154]:
```
0          40.108910
1          39.865420
2          39.102660
3          41.062130
4          39.172393
               ...
2845337    34.002480
2845338    32.766960
2845339    33.775450
2845340    33.992460
2845341    34.133930
Name: Start_Lat, Length: 2845342, dtype: float64
```

In [155…
```
USpro_dataset.Start_Lng
```

Out[155]:
```
0          -83.092860
1          -84.062800
2          -84.524680
3          -81.537840
4          -84.492792
              ...
2845337   -117.379360
2845338   -117.148060
2845339   -117.847790
2845340   -118.403020
2845341   -117.230920
Name: Start_Lng, Length: 2845342, dtype: float64
```

In [156…
```
sns.scatterplot(x=USpro_dataset.Start_Lng,y=USpro_dataset.Start_Lat)
```

Out[156]:
```
<AxesSubplot:xlabel='Start_Lng', ylabel='Start_Lat'>
```

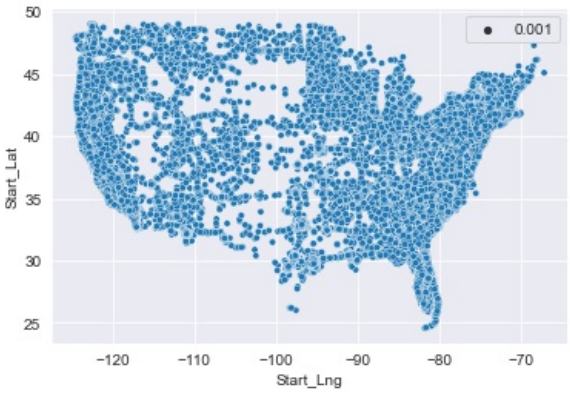


reduce the point size (using 10 percent)

In [162…
```
sample_USpro_dataset=USpro_dataset.sample(int(0.1* len(USpro_dataset)))
```

In [167…
```
sns.scatterplot(x=sample_USpro_dataset.Start_Lng,y=sample_USpro_dataset.Start_Lat,size=0.001)
```

Out[167]:
```
<AxesSubplot:xlabel='Start_Lng', ylabel='Start_Lat'>
```



import scatter plot on the map

In [175…
```
pip install folium
```

```
Collecting foliumNote: you may need to restart the kernel to use updated packages.

  Downloading folium-0.13.0-py2.py3-none-any.whl (96 kB)
Requirement already satisfied: numpy in c:\users\prade\anaconda3\lib\site-packages (from folium) (1.21.5)
Collecting branca>=0.3.0
  Downloading branca-0.6.0-py3-none-any.whl (24 kB)
Requirement already satisfied: requests in c:\users\prade\anaconda3\lib\site-packages (from folium) (2.27.1)
Requirement already satisfied: jinja2>=2.9 in c:\users\prade\anaconda3\lib\site-packages (from folium) (2.11.3)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\prade\anaconda3\lib\site-packages (from jinja2>=2.9
->folium) (2.0.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\prade\anaconda3\lib\site-packages (from requests-
>folium) (2021.10.8)
Requirement already satisfied: idna<4,>=2.5 in c:\users\prade\anaconda3\lib\site-packages (from requests->foliu
m) (3.3)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\prade\anaconda3\lib\site-packages (from reques
ts->folium) (1.26.9)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\prade\anaconda3\lib\site-packages (from re
quests->folium) (2.0.4)
Installing collected packages: branca, folium
Successfully installed branca-0.6.0 folium-0.13.0
```

In [176…
```
import folium
```

In [178…
```
lat,lon= USpro_dataset.Start_Lat[0],USpro_dataset.Start_Lng[0]
lat,lon
```

Out[178]:
```
(40.10891, -83.09286)
```

heat map is used

In [189…
```
from folium.plugins import HeatMap
```

In [190…
```
map = folium.Map()
HeatMap(zip(list(USpro_dataset.Start_Lat),list(USpro_dataset.Start_Lng))).add_to(map)
map
```

Out[190]:
```
Make this Notebook Trusted to load map: File -> Trust Notebook
```

In [188…
```
list(zip(list(USpro_dataset.Start_Lat),list(USpro_dataset.Start_Lng)))
```

Out[188]:
```
[(40.10891, -83.09286),
 (39.86542, -84.0628),
 (39.10266, -84.52468),
 (41.06213, -81.53784),
 (39.172393, -84.49279200000002),
 (39.06324, -84.03243),
 (39.77565, -84.18603),
 (41.37531, -81.82016999999998),
 (40.702247, -84.075887),
 (40.10931, -82.96849),
 (39.19288, -84.47723),
 (39.13877, -84.53394),
 (41.4739, -81.704233),
 (39.582242, -83.677814),
 (40.151785, -81.312635),
 (40.151747, -81.312682),
 (39.97241, -82.84695),
 (39.9838, -82.85656999999998),
 (40.02664, -82.9944),
 (41.679361, -83.573037),
 (40.99613, -85.26613),
 (39.0381, -84.59243000000002),
 (40.03386, -82.99601),
 (39.85798, -84.28181),
 (39.15267, -84.5395),
 (39.30732, -85.95982),
 (39.77572, -84.04815),
 (39.97643, -83.12306),
 (39.58595, -85.82518),
 (39.3638, -85.51659000000002),
 (39.45132, -85.62516),
 (39.75067, -84.14148),
 (39.2675, -81.49929),
 (39.86025, -85.96623000000002),
 (41.95677, -83.67214),
 (38.27401, -85.74844),
 (41.95677, -83.67214),
```

```
(40.52225, -80.06666),
(40.487814, -80.009439),
(40.156298, -83.018431),
(41.47461, -81.71181999999997),
(41.0618, -81.54608),
(40.45112, -85.15048),
(40.35429, -85.14993),
(39.75855, -85.13715),
(39.74408, -85.13749),
(39.965148, -83.020499),
(40.72813, -84.78965),
(39.85223, -85.25846999999997),
(41.46747, -81.75909),
(41.83193, -80.10143000000002),
(39.744287, -84.204939),
(39.71548, -84.22033),
(39.7504, -84.20561),
(41.72316, -84.96444),
(40.04376, -82.99708000000003),
(38.1781, -85.71946),
(38.18577, -85.80678),
(38.27191, -85.80838),
(39.923905, -82.87008),
(41.035566, -81.569917),
(39.9239, -83.68767),
(41.851914, -80.175232),
(39.93849, -82.84849),
(38.96943, -80.1096),
(41.66805, -83.57063000000002),
(41.48339, -81.66297),
(41.50127, -81.4804),
(39.789093, -82.989106),
(41.67073, -81.24561),
(41.70846, -81.17636),
(41.44246, -81.78485),
(39.77128, -84.1923),
(41.50499, -81.47417),
(40.109653, -80.2029),
(38.38852, -81.7687),
(41.03572, -81.57809),
(41.47487, -81.72095),
(41.62845, -84.80559000000002),
(41.62894, -84.80373),
(41.16102, -81.78573),
(41.628232, -84.808858),
(41.62986, -84.76619000000002),
(39.74729, -84.21426),
(41.42099, -81.69051999999998),
(41.42318, -81.84674),
(38.79691, -84.48273),
(38.80878, -84.49638),
(38.30155, -85.85499),
(38.33667, -81.65623000000002),
(38.33614, -81.65623000000002),
(41.0961, -81.58593),
(41.12155, -85.18715999999998),
(38.829993, -80.667067),
(40.05642, -83.03097),
(41.37717, -81.5139),
(39.17397, -84.49031),
(39.75513, -84.16614),
(39.17397, -84.49031),
(41.12624, -81.65299),
(38.33667, -81.65623000000002),
(41.10389, -81.5),
(39.16417, -84.45275),
(40.11184, -83.00883),
(38.31922, -85.75301999999998),
(39.09881, -84.50801),
(41.473859, -81.708409),
(41.039988, -81.565332),
(41.73565, -84.99082),
(39.26566, -84.43965),
(41.46981, -81.81104),
(41.25666, -81.63157),
(40.58919, -80.09885),
(41.06356, -81.49078),
(41.06218, -81.50595),
(39.17736, -84.4873),
(38.33956, -81.60991999999997),
(41.47889, -81.6934),
(40.58919, -80.09885),
(41.41119000000001, -81.64414000000002),
(39.24907, -84.44561999999998),
(39.28775, -84.40073000000002),
(41.47379, -81.89631),
(41.54941, -81.60676),
(39.03347, -84.603),
(39.74903, -84.20106),
```

```
(39.89473, -83.044),
(39.19641, -84.47331),
(39.22461, -84.45364000000002),
(39.15306, -84.44283),
(39.97262, -82.98368),
(40.535447, -80.03596),
(40.529455, -80.03506999999998),
(41.49647, -81.67733),
(41.0634, -81.49079),
(39.74284, -84.20479),
(41.46822, -81.9476),
(41.92479, -83.36026),
(41.184738, -80.432713),
(40.27317, -85.5577),
(39.98264, -85.14301999999998),
(40.27317, -85.5577),
(40.42667, -85.54968000000002),
(39.97527, -85.14018),
(39.09889, -84.52118),
(38.88012, -84.62527),
(38.79256, -84.60423),
(39.74284, -84.20479),
(40.484222, -80.13755400000002),
(41.20104, -80.06859),
(39.05528, -84.39072),
(41.623325, -83.692226),
(40.53024, -80.00957),
(40.06527, -82.99591),
(40.43954, -80.00971),
(41.422004, -81.730092),
(39.3638, -85.51659000000002),
(39.82083, -85.7749),
(40.0366, -85.72561999999998),
(39.855691, -84.29079200000002),
(40.049152, -85.682047),
(39.83939, -85.55752),
(40.13386, -80.00666),
(40.43954, -80.00971),
(40.13386, -80.00666),
(40.13386, -80.00666),
(39.97859, -82.9763),
(39.95459, -83.04255),
(39.68291, -85.97372),
(38.49304, -85.77156),
(38.33428, -85.82177),
(39.99921, -83.02875),
(38.53844, -85.77702),
(38.1781, -85.71946),
(39.87433, -82.99882),
(41.39546, -85.32855),
(41.39469, -85.28893000000002),
(41.90473, -83.36986),
(40.77137000000001, -84.62203000000002),
(40.80059, -84.61374),
(39.87433, -82.99882),
(40.78593, -84.61787),
(40.80059, -84.61374),
(41.67487, -83.57266),
(41.47376, -81.86113),
(38.33465, -85.75389),
(39.14014, -84.53495),
(41.80143, -85.34888000000002),
(41.35761, -81.81988),
(39.34392, -84.39043000000002),
(39.816154, -83.18010500000003),
(38.824929, -85.47449499999998),
(38.82415, -85.63794),
(40.593244, -85.150017),
(40.600487, -85.150502),
(41.19362, -80.2374),
(40.55267, -85.14974000000002),
(40.56744000000001, -85.14993),
(40.14212, -81.55153),
(39.94397, -82.41127),
(39.96893, -83.11922),
(40.6126, -80.09494000000002),
(40.11184, -83.00883),
(41.06347, -81.50372),
(39.972931, -85.965783),
(39.95459, -83.04255),
(40.11184, -83.00883),
(39.97241, -82.84695),
(39.9912, -85.9188),
(41.34965, -81.51146),
(41.11987, -80.69004),
(41.11851, -80.69242),
(40.05642, -83.03097),
(38.117487, -85.760397),
(41.44134, -81.65055),
```

```
(38.61531, -80.76034),
(41.124052, -80.677589),
(39.95085, -82.94428),
(41.06257, -81.52228000000002),
(40.11212, -83.03905999999998),
(39.887033, -83.043909),
(40.04916, -83.03336),
(41.12906, -85.1612),
(41.17798, -85.10323000000002),
(38.08783, -83.90263),
(41.63316, -83.54203000000003),
(39.149639, -84.539381),
(40.9605, -85.3528),
(40.96051, -85.34522),
(41.08204, -80.64846999999997),
(41.61791, -83.54106999999998),
(38.41592, -82.3465),
(41.67571, -83.69385),
(39.08272, -84.52264),
(39.280947, -80.278075),
(38.36387, -81.73867),
(40.0187, -82.90465999999998),
(41.53757, -81.64083000000002),
(41.16522, -81.47454),
(39.52352, -82.02360999999998),
(39.517216, -82.025802),
(40.322818, -83.071034),
(41.73565, -84.99082),
(39.14648, -84.45455),
(39.82122, -85.76482),
(41.11889, -81.65406999999998),
(40.121557, -83.151457),
(40.11017, -83.15876999999998),
(41.57777, -81.54987),
(40.121557, -83.151457),
(40.11017, -83.15876999999998),
(41.04592, -81.69463),
(41.42099, -81.69051999999998),
(39.27486, -84.346),
(39.28266, -84.56742),
(39.9467, -82.22232),
(39.17397, -84.49031),
(39.17736, -84.4873),
(39.9467, -82.22232),
(39.9467, -82.22232),
(41.68518, -83.5678),
(41.66572, -83.5652),
(39.27125, -84.35176),
(41.47395, -81.69931),
(39.98505, -80.74006),
(39.95813, -80.76219),
(41.56544, -81.5835),
(38.32587, -83.11389),
(38.25869, -85.7622),
(38.29322, -85.654625),
(39.78609, -83.24606999999997),
(41.53968, -81.62795),
(39.08927, -84.5227),
(38.25922, -85.76632),
(41.55264, -81.60032),
(39.02187, -84.50492),
(41.06755, -81.57421),
(40.49223, -80.01029),
(41.48836, -81.66471),
(39.95085, -82.94428),
(41.42063, -81.80859),
(41.06755, -81.57421),
(40.82949, -81.39732),
(41.40091, -81.81724),
(41.13355, -85.13584),
(38.870386, -84.624429),
(41.47974, -81.6667),
(40.30095, -80.16651999999998),
(39.11543, -84.50003000000002),
(41.1419, -81.47694),
(39.14917, -84.44711),
(39.81979000000001, -84.18908),
(39.84509, -84.1899),
(41.47461, -81.71181999999997),
(39.17304, -84.47589),
(39.1021, -84.49754),
(41.49873, -81.67260999999998),
(39.12627, -84.53509),
(41.47394600000001, -81.699011),
(39.9745, -83.09654),
(39.85911, -84.2808),
(39.24943, -84.36541),
(38.31179, -85.58773000000002),
(39.27486, -84.346),
```

```
(39.29287, -84.31981),
(39.15267, -84.5395),
(41.4675, -81.49014),
(39.26661, -84.35605),
(39.27125, -84.35176),
(38.20674, -85.74846),
(40.13804, -82.97112),
(41.47965, -81.66758),
(39.08887, -84.52271),
(41.6882, -83.64064),
(40.06527, -82.99591),
(39.86536, -84.05238),
(38.420223, -82.293065),
(41.47414000000001, -81.69893),
(41.47542, -81.68721),
(40.07297, -83.1347),
(40.05609000000001, -80.679211),
(38.19105, -85.71902),
(38.25454000000001, -85.71969),
(39.171694, -84.494399),
(41.06239, -81.52229),
(39.259254, -84.440978),
(39.905178, -82.895471),
(40.0988, -83.13331),
(41.33991, -81.81739),
(40.0999, -82.92089),
(39.4728, -80.11963),
(41.46631, -81.77065999999998),
(38.34109, -81.70017),
(41.160072, -85.232647),
(41.163626, -85.24482900000002),
(39.14014, -84.53495),
(40.94537, -81.15473),
(41.68591, -83.911307),
(41.704523, -83.9136),
(40.945772, -81.17219399999998),
(38.82445, -82.98299),
(38.84111, -82.98432),
(39.00079, -84.42043000000002),
(39.96071, -85.35383),
(39.2675, -81.49929),
(40.27317, -85.5577),
(40.42667, -85.54968000000002),
(38.130782, -81.388878),
(38.545195, -85.778613),
(38.362427, -81.714434),
(38.39115, -81.58664),
(40.08978, -82.98725999999998),
(40.0122, -82.99188000000002),
(41.68918, -83.53681999999998),
(39.3781, -84.365),
(39.11919, -84.53539),
(39.05029, -83.77673),
(39.04827, -83.77199),
(40.0717, -82.90821),
(39.09655, -84.48312),
(40.43077, -80.02638),
(41.443708, -85.19285),
(41.438119, -85.15969399999999),
(41.46747, -81.75909),
(39.044435, -84.465864),
(39.22461, -84.45364000000002),
(41.104217, -81.499987),
(41.1003, -81.4999),
(39.1021, -84.49754),
(41.46375, -81.69353000000002),
(41.52134, -83.45642),
(41.52549000000001, -83.46344),
(41.43166, -81.68160999999998),
(38.22363, -85.50532),
(40.85466, -81.41646999999998),
(38.22272, -85.50111),
(39.12424, -84.53501),
(39.09858, -84.51974),
(39.63551, -84.19519),
(40.39715, -80.10368000000003),
(41.56709, -81.57816),
(41.18225, -84.94275999999998),
(39.10148, -84.52341),
(41.55245, -83.60061999999998),
(38.52006, -81.35221),
(41.3513, -83.62276),
(39.25015, -81.30566),
(41.345031, -82.285379),
(41.351335, -82.26144599999998),
(38.38457, -81.804429),
(40.41528, -80.01234000000002),
(40.03598, -83.12479),
(41.03669, -81.50492),
```

```
(41.68971, -83.53045999999998),
(39.587784, -83.723939),
(39.626237, -83.73105),
(41.06243, -81.49904000000002),
(39.97262, -82.98368),
(39.93658, -83.00941999999998),
(41.208579, -82.807829),
(41.4051, -81.8184),
(39.29035, -84.39078),
(39.90524, -82.89541),
(40.04376, -82.99708000000003),
(39.98793, -82.98568),
(41.874915, -85.409527),
(41.890234, -85.409642),
(39.9335, -82.78939),
(40.04916, -83.03336),
(39.99493, -82.93332),
(39.83727, -83.09326999999998),
(40.98954000000001, -81.49372),
(38.53844, -85.77702),
(41.10389, -81.5),
(38.577894, -85.77896),
(41.42099, -81.69051999999998),
(38.58054600000001, -85.77866),
(39.833935, -84.061832),
(38.30344, -85.88893),
(39.93654, -83.40048),
(39.95612, -83.37499),
(39.17766, -84.39780999999998),
(40.02155, -83.0354),
(39.97518, -85.63374),
(39.94576, -85.63454),
(40.27744000000001, -85.72518000000002),
(39.17304, -84.47589),
(41.68648, -83.5669),
(41.66267, -85.03312),
(41.68976, -83.666),
(39.94594, -82.9432),
(41.4758, -81.66036),
(41.628312, -84.812135),
(39.96724, -83.02459),
(41.96429000000001, -83.35018000000002),
(41.282993, -83.638931),
(41.63126, -83.48336),
(41.427584, -85.8495),
(41.471378, -85.839527),
(41.46747, -81.75909),
(39.18364, -84.48373000000002),
(41.94495, -80.40234),
(39.53078, -84.310264),
(41.438774, -81.80341999999997),
(39.9838, -82.85656999999998),
(40.44413400000001, -80.028261),
(39.97727, -83.15445),
(39.9745, -83.09654),
(39.97865, -83.11698),
(39.97327, -83.12002),
(39.22571, -84.36832),
(39.986342, -83.118881),
(39.73629, -84.20489),
(41.4051, -81.8184),
(41.41101, -81.66634),
(39.24464, -84.44835),
(40.11212, -83.03905999999998),
(40.30095, -80.16651999999998),
(41.49243, -81.67554),
(40.1827, -80.26148),
(39.95459, -83.04255),
(39.97527, -85.14018),
(39.98511, -85.14406),
(38.29514, -85.75655),
(39.93654, -83.40048),
(39.11543, -84.50003000000002),
(41.030684, -81.885739),
(41.43166, -81.68160999999998),
(40.6664, -80.227514),
(40.67137, -80.22581),
(41.305199, -81.438626),
(40.354252, -85.843032),
(40.402251, -85.842969),
(41.26752000000001, -85.85673),
(39.97307, -82.98405),
(39.9672, -81.28699999999998),
(39.96392, -81.27197),
(41.52631, -83.62123000000003),
(39.93292, -82.83025),
(39.08272, -84.52264),
(39.8552, -84.32502),
(41.65404, -85.70526),
```

(41.39465, -81.65326),
(41.06211, -81.53053),
(41.09598, -81.49976),
(41.85222, -85.67843),
(38.148808, -80.929665),
(41.52447, -85.57791),
(41.5105, -85.57781),
(39.90524, -82.89541),
(41.23598, -81.49325999999998),
(38.70101, -80.66483000000002),
(39.22498, -84.38094),
(40.776023, -80.12718100000002),
(38.86027, -80.65729),
(39.88203, -83.04748000000002),
(39.89253400000001, -83.038786),
(39.89698, -83.03428000000002),
(41.67439, -83.693727),
(41.12999, -85.15503000000002),
(39.09889, -84.52118),
(41.52631, -83.62123000000003),
(39.088552, -84.522724),
(39.95423, -82.98879000000002),
(39.21018, -84.46345),
(39.95369, -82.96403000000002),
(40.40934, -80.19247),
(40.44126, -80.19314),
(39.16161, -84.51375),
(39.16161, -84.51375),
(39.15141, -84.54061),
(39.75855, -85.13715),
(41.67571, -83.69385),
(41.67571, -83.69385),
(39.28805, -84.37462),
(41.53618, -83.6176),
(39.07202, -84.51907),
(38.259133, -85.752672),
(40.44425, -80.02821),
(40.44278, -80.02468),
(38.30155, -85.85499),
(40.35405, -80.13756),
(40.35449000000001, -80.13835999999998),
(38.30344, -85.88893),
(41.08789, -83.66011),
(41.02454, -80.92163000000002),
(41.02413, -80.90579),
(39.08168, -84.7069),
(38.18539000000001, -85.81793),
(39.27734, -84.86438000000003),
(39.16437, -84.43731),
(39.748585, -84.226084),
(39.4728, -80.11963),
(41.187807, -80.149019),
(41.19242, -80.15860500000002),
(38.233483, -85.62671800000003),
(38.33954, -81.609895),
(40.07297, -83.1347),
(40.08402, -83.13046),
(38.34631, -83.40449),
(38.35382, -83.40553),
(39.04416, -84.46656),
(38.41807, -81.54525),
(38.309401, -85.847643),
(41.64679, -83.54913),
(38.198748, -85.733953),
(39.97401, -82.99355),
(39.17397, -84.49031),
(41.68843, -83.56258000000003),
(41.67904, -83.5732),
(39.25249, -84.36493),
(38.310736, -85.590395),
(41.46941, -81.80535),
(38.20674, -85.74846),
(39.226317, -84.37107900000002),
(41.68721, -83.55579),
(40.09775, -84.22926),
(41.42147, -81.70682),
(38.29414600000001, -85.756847),
(39.17304, -84.47589),
(38.15982, -84.67308),
(39.11675, -84.53355),
(39.24943, -84.36541),
(41.06146, -81.5049),
(39.18364, -84.48373000000002),
(39.98793, -82.98568),
(39.95971, -83.04811),
(39.1479, -84.53929000000002),
(39.89555, -83.84997),
(39.99226, -82.98517),
(39.40983, -82.97548),

```
(40.6126, -80.09494000000002),
(39.22479000000001, -84.27783000000002),
(39.11675, -84.53355),
(41.43869, -81.80554000000002),
(40.016819, -82.90411),
(39.09757000000001, -84.51666),
(41.06247000000001, -81.52405),
(39.13219, -84.49464),
(39.097654, -84.516019),
(41.34965, -81.51146),
(41.37717, -81.5139),
(40.12589000000001, -80.13661),
(40.13189000000001, -80.132407),
(40.00507, -83.1186),
(39.43794000000001, -82.53893000000002),
(39.12627, -84.53509),
(38.07759, -84.45711999999997),
(40.09079000000001, -82.90783),
(39.64731, -85.91696999999998),
(39.08927, -84.5227),
(38.26109, -85.73695),
(38.19785, -85.70445),
(39.05654000000001, -84.54283000000002),
(41.42413, -81.64519),
(39.28638, -84.42706),
(41.37499, -83.61635),
(41.06228, -81.50986999999998),
(41.53907, -81.63354),
(40.04916, -83.03336),
(41.13613, -81.72196),
(39.09891, -84.49181999999998),
(39.96268, -82.84305),
(39.17397, -84.49031),
(39.16911, -84.42104),
(39.003821, -80.391285),
(39.17736, -84.4873),
(39.14316, -84.47069),
(39.2696, -84.34894),
(40.521732, -81.64755),
(40.513361, -81.651599),
(41.52631, -83.62123000000003),
(38.992607, -80.195782),
(38.839151, -82.152877),
(40.27675, -85.49916),
(40.52168, -80.13163),
(39.83711, -84.54392),
(39.02296, -84.51784),
(38.455787, -81.88205),
(39.02397, -84.48039),
(38.22433, -85.751),
(39.05973, -84.42438),
(39.27734, -84.86438000000003),
(41.47965, -81.66758),
(39.02257, -84.48631),
(40.44846, -80.00755),
(39.09891, -84.28556999999998),
(41.06026, -81.55485999999998),
(41.802758, -83.684687),
(38.0791, -83.94591),
(38.01846, -84.15159),
(38.30344, -85.88893),
(38.84109, -85.81618),
(41.14181, -80.72015999999998),
(39.14648, -84.45455),
(41.04263, -81.50543),
(38.21089, -85.82606),
(39.75067, -84.14148),
(38.28581, -85.50792),
(39.15273300000001, -84.541403),
(38.23585, -85.658849),
(38.20249, -85.68697),
(40.53491, -83.10678),
(40.52035, -83.10043),
(39.12965, -84.49555),
(38.22811, -85.7514),
(40.41233, -80.07815),
(39.7504, -84.20561),
(39.08272, -84.52264),
(39.781457, -85.839651),
(39.781457, -85.839651),
(41.06076, -81.45923),
(39.688148, -83.88431),
(39.693164, -83.882745),
(39.9488, -83.02845),
(41.11584000000001, -81.61428000000002),
(39.797697, -84.189677),
(39.20326, -85.95763000000002),
(39.24464, -84.44835),
(39.73733, -84.25733000000002),
```

```
(39.969853, -85.972976),
(41.11889, -81.65406999999998),
(39.13219, -84.49464),
(41.458, -83.62201999999998),
(40.09268, -83.03726),
(40.066965, -83.289888),
(40.045608, -83.29660799999998),
(39.97307, -82.98405),
(39.98248, -85.123922),
(39.981095, -85.11027299999998),
(40.43021, -80.02554),
(40.43219000000001, -80.02781999999998),
(39.976567, -83.128639),
(41.256658, -85.363439),
(41.37499, -83.61635),
(41.252653, -85.347328),
(39.93658, -83.00941999999998),
(39.14014, -84.53495),
(40.153014, -82.970316),
(39.90524, -82.89541),
(39.13219, -84.49464),
(40.42581, -80.4294),
(41.08789, -83.65977),
(39.9488, -83.02845),
(41.47585, -81.87980999999998),
(39.1079, -84.50284),
(38.340803, -81.680728),
(40.96473, -85.28862),
(38.441, -82.12679),
(38.4413, -82.01529000000002),
(38.441, -82.12679),
(38.18977, -83.48379),
(41.194131, -80.201982),
(40.09601, -83.13626),
(41.23632, -83.65401999999997),
(38.1737, -84.81473000000003),
(38.19016, -85.7616),
(39.25249, -84.36493),
(39.137217, -84.04718299999998),
(39.141454, -84.016543),
(38.344424, -81.613662),
(41.93113, -83.35857),
(40.89268, -84.61808),
(40.46013, -80.19126),
(41.119342, -80.698025),
(41.117872, -80.693898),
(39.15267, -84.5395),
(38.27362, -85.80906),
(41.458, -83.62201999999998),
(39.35655, -84.26419),
(38.4062, -81.54054000000002),
(41.02525900000001, -81.5042),
(38.27362, -85.80906),
(38.11969000000001, -85.77235),
(38.275406, -85.81314300000003),
(38.27508, -85.81517099999998),
(41.39769, -81.939),
(39.46513, -80.13559000000002),
(38.217994, -81.426024),
(38.215401, -81.42437199999998),
(39.042851, -84.612615),
(38.26109, -85.73695),
(39.05654000000001, -84.54283000000002),
(41.840758, -84.361657),
(41.841123, -84.361652),
(41.63316, -83.54203000000003),
(40.02874, -81.04368000000002),
(41.43891, -81.80258),
(41.0811, -81.50348000000002),
(38.35575, -81.65420999999998),
(39.94585, -82.94293),
(41.69017, -83.5181),
(40.11212, -83.03905999999998),
(41.50259000000001, -81.66974),
(39.989994, -83.02512),
(41.480458, -81.692932),
(39.22461, -84.45364000000002),
(40.04376, -82.99708000000003),
(39.93621, -82.88318000000002),
(39.167788, -82.524172),
(39.20655, -82.48212),
(41.36576, -81.51602),
(39.977322, -81.183327),
(39.961933, -81.175436),
(39.68292, -84.23004),
(40.02664, -82.9944),
(38.583372, -81.994824),
(38.58119600000001, -82.004356),
(40.07297, -83.1347),
```

(41.58689, -83.56234),
(38.31714, -85.57677),
(40.07297, -83.1347),
(39.07202, -84.51907),
(41.46747, -81.75909),
(40.82645, -81.39675),
(40.82645, -81.39675),
(39.05654000000001, -84.54283000000002),
(39.06272, -84.65728),
(39.03368, -84.59527),
(40.02121, -82.99503),
(39.866695, -84.9305),
(39.05654000000001, -84.54283000000002),
(39.03232, -84.59137),
(41.04058, -81.61256999999998),
(40.98884, -83.64875),
(39.95658, -83.37075),
(39.89857, -83.72139),
(39.896448, -83.72312099999998),
(41.47965, -81.66758),
(39.71548, -84.22033),
(40.0389, -85.71828000000002),
(41.17516, -81.23116999999998),
(41.15107, -81.21375),
(40.0389, -85.71828000000002),
(39.99336, -85.83861999999998),
(40.46013, -80.19126),
(41.68976, -83.666),
(39.97326, -83.00690999999998),
(39.48572, -83.94191),
(39.83355, -85.99635),
(39.79721, -85.97153),
(39.98056, -83.11924),
(41.458, -83.62243000000002),
(39.905352, -82.995352),
(39.99921, -83.02875),
(40.869049, -81.95845600000001),
(41.419021, -81.519515),
(39.34392, -84.39043000000002),
(39.9454, -82.60271),
(38.353957, -81.73513100000002),
(38.354505, -81.731585),
(41.501623, -81.479519),
(39.86477, -83.99871),
(38.737813, -85.250959),
(40.00507, -83.1186),
(39.91978, -82.93149),
(39.09757000000001, -84.51666),
(41.07995, -81.50389),
(41.63316, -83.54203000000003),
(39.82529, -83.03175999999998),
(39.83209, -82.99884),
(39.83208, -83.000447),
(38.416631, -84.861968),
(40.55549, -80.11634000000002),
(39.17736, -84.4873),
(41.484589, -81.691778),
(41.22734000000001, -81.62705),
(39.04695, -84.57498000000002),
(39.25595, -84.44287),
(39.8888, -82.88181999999998),
(39.74729, -84.21426),
(39.97415, -83.09486),
(39.08272, -84.52264),
(39.949808, -83.02289300000002),
(39.95141, -83.0137),
(38.7056, -80.66107),
(39.1479, -84.53929000000002),
(41.7573, -83.48075),
(39.98946, -85.9299),
(39.98894, -83.025012),
(41.110996, -80.82384),
(41.68685900000001, -83.554005),
(41.53907, -81.63354),
(39.300562, -84.518869),
(41.202543, -85.7011),
(39.19288, -84.47723),
(40.07297, -83.1347),
(39.140991, -84.483618),
(41.121599, -80.789503),
(41.31438, -81.51286),
(39.85922, -84.27762),
(40.42005, -80.03531),
(39.9589, -82.98257),
(39.89087, -83.85382),
(39.89862, -83.85353),
(40.03938, -83.050567),
(41.41218, -81.61371),
(38.306159, -80.833765),

```
(40.104, -82.94154),
(38.24178, -85.62212),
(41.55264, -81.60032),
(39.99226, -82.98517),
(38.31553, -85.58313000000003),
(38.400331, -85.371049),
(38.31251, -85.57741999999998),
(38.400331, -85.371049),
(39.88938, -82.88316),
(38.15873, -80.90504),
(39.9745, -83.09654),
(41.47487, -81.72095),
(39.26931, -84.4307),
(38.2201, -85.82223),
(41.48339, -81.66297),
(39.16161, -84.51375),
(41.42734, -82.08322),
(41.52631, -83.62123000000003),
(39.91732, -83.012335),
(41.60867, -81.48891),
(41.20554, -85.17116),
(41.507189, -81.67124),
(39.97415, -83.09486),
(39.97043, -83.01961999999997),
(39.93849, -82.84849),
(39.89698, -83.03428000000002),
(39.71452, -84.22281),
(40.11468, -83.01644),
(38.34244, -85.80447),
(41.47036, -81.65984),
(38.30056, -85.84703),
(39.169389, -84.50389),
(39.936838, -83.042661),
(39.937567, -83.04209200000003),
(39.9425, -82.53536),
(39.27125, -84.35176),
(41.398877, -83.616108),
(39.24422, -84.30224),
(40.412997, -80.07973299999998),
(39.021521, -84.50784300000002),
(41.12367, -80.76913),
(41.073876, -83.66091999999998),
(39.01992, -83.65886),
(38.97914, -83.65465),
(40.09927, -83.15793000000002),
(39.93658, -83.00941999999998),
(39.11543, -84.50003000000002),
(41.474552, -81.711522),
(39.78517, -84.18457),
(41.47974, -81.6667),
(38.27719, -85.69383),
(38.495124, -85.771118),
(39.73277, -84.20522),
(41.458, -83.62243000000002),
(39.045381, -84.462312),
(41.526, -81.65974),
(40.78725, -81.38592),
(38.3657, -85.75495),
(39.02575, -84.61272),
(40.90923, -81.79765),
(40.91649, -81.78869),
(39.10051, -80.39199),
(40.082643, -82.91367199999998),
(41.47162, -81.90626),
(40.08155, -82.91859000000002),
(40.07976, -82.92689),
(38.27511, -85.81238),
(39.02908, -84.53565),
(39.04455, -84.57798000000003),
(39.83026, -81.45434),
(39.81436, -81.46898),
(38.41336, -82.3483),
(38.20674, -85.74846),
(38.10937, -85.70123000000002),
(40.26758, -82.92745),
(38.22285, -85.57436),
(40.43827, -80.00527),
(39.08272, -84.52264),
(40.046599, -85.99359),
(41.67904, -83.5732),
(40.52225, -80.06666),
(41.041763, -81.3932),
(39.02541, -85.88784),
(39.887572, -84.047499),
(39.90803, -84.03969000000002),
(39.8181, -81.45925),
(39.8181, -81.45925),
(39.81447, -81.46769),
(41.47307, -80.36586),
```

```
(41.460561, -80.366376),
(39.75022, -84.19668),
(39.16161, -84.51375),
(39.82329, -84.166569),
(39.818751, -84.167637),
(39.15267, -84.5395),
(40.480209, -85.719799),
(39.02676, -84.61421),
(40.480209, -85.719799),
(40.48013, -85.72942900000002),
(38.24863, -85.70467),
(39.24907, -84.44561999999998),
(39.18364, -84.48373000000002),
(41.420898, -81.69322),
(39.18976, -84.26285),
(40.00921, -83.03149),
(41.12369, -80.75595),
(41.09705, -81.50012),
(41.03011, -81.40292099999998),
(39.9533, -83.0032),
(39.91404, -83.01729),
(39.11811, -84.49975),
(39.73277, -84.20522),
(41.47379, -81.69592),
(41.47395, -81.69931),
(39.62334, -81.83577),
(39.98233, -82.98449000000002),
(40.15106, -80.03243),
(41.66332, -83.56385999999998),
(39.35245, -84.37507),
(41.67178, -83.6939),
(38.38018, -82.60993),
(39.03347, -84.603),
(40.42116, -80.0438),
(41.341147, -83.347543),
(41.34131, -83.376561),
(40.26758, -82.92745),
(40.432459, -80.023538),
(39.9912, -85.9188),
(39.05654000000001, -84.54283000000002),
(38.39328, -85.76223),
(39.97165800000001, -80.017133),
(39.98381, -80.01011),
(39.84457, -85.51639399999998),
(38.399528, -85.76430500000002),
(38.38981, -81.76937),
(41.47965, -81.66758),
(39.97859, -82.9763),
(40.68109000000001, -80.24616999999998),
(41.47487, -81.72095),
(40.68109000000001, -80.24616999999998),
(40.1412, -82.97121),
(39.85942, -84.27778),
(39.10838, -84.50296999999998),
(39.32662, -84.42002),
(39.24907, -84.44561999999998),
(41.11959, -81.64568),
(39.04455, -84.57798000000003),
(38.26109, -85.73695),
(38.21911, -85.50582),
(39.94976, -83.04032),
(38.82584, -120.029214),
(37.358209, -121.840017),
(37.881943, -122.307987),
(37.881038, -122.307788),
(38.518811, -121.101664),
(38.518811, -121.101664),
(36.9903, -119.71146),
(37.42592, -122.09879),
(37.75745, -122.21131),
(37.31648, -121.96746),
(37.44415, -122.2688),
(37.71981, -121.65943),
...]
```

In [ ]:

In [ ]:

In [ ]:

SUMMARY AND CONCLUSION

Insights:

1. no data for new york
2. the number of accident per city decreses exponentially.
3. less than 5% of cities have more than 1000 yearly accidents.
4. over 12k cities have reported just one accident(need to investicate).
5. when is accidents per unit of traffic the highest.

In [ ]: