Police Dataset

Here, The data from a police check post is given.

I'm going to analyze this dataset using the Pandas DataFrame

In [1]:
```
import pandas as pd
```

In [3]:
```
df=pd.read_csv(r"C:\Users\Sathiyamurthy\Downloads\Police Data.csv")
```

In [4]:
```
df
```

Out[4]:

	stop_date	stop_time	country_name	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search_conducted	search_
0	1/2/2005	1:55	NaN	M	1985.0	20.0	White	Speeding	Speeding	False	
1	1/18/2005	8:15	NaN	M	1965.0	40.0	White	Speeding	Speeding	False	
2	1/23/2005	23:15	NaN	M	1972.0	33.0	White	Speeding	Speeding	False	
3	2/20/2005	17:15	NaN	M	1986.0	19.0	White	Call for Service	Other	False	
4	3/14/2005	10:00	NaN	F	1984.0	21.0	White	Speeding	Speeding	False	
...	...	...	...	...	...	...	...	...	...	...	
65530	12/6/2012	17:54	NaN	F	1987.0	25.0	White	Speeding	Speeding	False	
65531	12/6/2012	22:22	NaN	M	1954.0	58.0	White	Speeding	Speeding	False	
65532	12/6/2012	23:20	NaN	M	1985.0	27.0	Black	Equipment/Inspection Violation	Equipment	False	
65533	12/7/2012	0:23	NaN	NaN	NaN	NaN	NaN	NaN	NaN	False	
65534	12/7/2012	0:30	NaN	F	1985.0	27.0	White	Speeding	Speeding	False	

65535 rows × 15 columns

i)Remove the missing column that only contains missing values

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
stop_date                 0
stop_time                 0
country_name          65535
driver_gender          4061
driver_age_raw         4054
driver_age             4307
driver_race            4060
violation_raw          4060
violation              4060
search_conducted          0
search_type           63056
stop_outcome           4060
is_arrested            4060
stop_duration          4060
drugs_related_stop        0
dtype: int64
```

In [6]:
```
df.drop(columns='country_name', inplace=True)
```

In [7]:
```
df
```

Out[7]:

	stop_date	stop_time	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search_conducted	search_type	stop_outc
0	1/2/2005	1:55	M	1985.0	20.0	White	Speeding	Speeding	False	NaN	Cit
1	1/18/2005	8:15	M	1965.0	40.0	White	Speeding	Speeding	False	NaN	Cit
2	1/23/2005	23:15	M	1972.0	33.0	White	Speeding	Speeding	False	NaN	Cit
3	2/20/2005	17:15	M	1986.0	19.0	White	Call for Service	Other	False	NaN	Arrest D
4	3/14/2005	10:00	F	1984.0	21.0	White	Speeding	Speeding	False	NaN	Cit
...	...	...	...	...	...	...	...	...	...	...	
65530	12/6/2012	17:54	F	1987.0	25.0	White	Speeding	Speeding	False	NaN	Cit
65531	12/6/2012	22:22	M	1954.0	58.0	White	Speeding	Speeding	False	NaN	Wa
65532	12/6/2012	23:20	M	1985.0	27.0	Black	Equipment/Inspection Violation	Equipment	False	NaN	Cit
65533	12/7/2012	0:23	NaN	NaN	NaN	NaN	NaN	NaN	False	NaN	
65534	12/7/2012	0:30	F	1985.0	27.0	White	Speeding	Speeding	False	NaN	Cit

65535 rows × 14 columns

ii) For Speeding, were Men or Women stopped more often?

In [8]:
```
df.head()
```

Out[8]:

	stop_date	stop_time	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search_conducted	search_type	stop_outcome	is_arr
0	1/2/2005	1:55	M	1985.0	20.0	White	Speeding	Speeding	False	NaN	Citation	
1	1/18/2005	8:15	M	1965.0	40.0	White	Speeding	Speeding	False	NaN	Citation	
2	1/23/2005	23:15	M	1972.0	33.0	White	Speeding	Speeding	False	NaN	Citation	
3	2/20/2005	17:15	M	1986.0	19.0	White	Call for Service	Other	False	NaN	Arrest Driver	
4	3/14/2005	10:00	F	1984.0	21.0	White	Speeding	Speeding	False	NaN	Citation	

In [9]:
```
df[df.violation == 'Speeding'].driver_gender.value_counts()
```

Out[9]:
```
M    25517
F    11686
Name: driver_gender, dtype: int64
```

iii) Does gender affect who gets searched during a stop?

In [10]:
```
df.groupby('driver_gender').search_conducted.sum()
```

Out[10]:
```
driver_gender
F     366.0
M    2113.0
Name: search_conducted, dtype: float64
```

In [11]:
```
df.search_conducted.value_counts()
```

Out[11]:
```
False    63056
True      2479
Name: search_conducted, dtype: int64
```

(mapping + data-type casting)

iv) what is the mean stop_duration?

In [12]:
```
df.stop_duration.value_counts()
```

Out[12]:
```
0-15 Min     47379
16-30 Min    11448
30+ Min       2647
2                1
Name: stop_duration, dtype: int64
```

In [14]:
```
df["stop_duration"]=df["stop_duration"].map({'0-15 Min' : 7.5,'16-30 Min' :24,'30+ Min' : 45})
```

In [15]:
```
df
```

Out[15]:

	stop_date	stop_time	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search_conducted	search_type	stop_outc
0	1/2/2005	1:55	M	1985.0	20.0	White	Speeding	Speeding	False	NaN	Cit
1	1/18/2005	8:15	M	1965.0	40.0	White	Speeding	Speeding	False	NaN	Cit
2	1/23/2005	23:15	M	1972.0	33.0	White	Speeding	Speeding	False	NaN	Cit
3	2/20/2005	17:15	M	1986.0	19.0	White	Call for Service	Other	False	NaN	Arrest D
4	3/14/2005	10:00	F	1984.0	21.0	White	Speeding	Speeding	False	NaN	Cit
...	...	...	...	...	...	...	...	...	...	...	
65530	12/6/2012	17:54	F	1987.0	25.0	White	Speeding	Speeding	False	NaN	Cit
65531	12/6/2012	22:22	M	1954.0	58.0	White	Speeding	Speeding	False	NaN	Wa
65532	12/6/2012	23:20	M	1985.0	27.0	Black	Equipment/Inspection Violation	Equipment	False	NaN	Cit
65533	12/7/2012	0:23	NaN	NaN	NaN	NaN	NaN	NaN	False	NaN	
65534	12/7/2012	0:30	F	1985.0	27.0	White	Speeding	Speeding	False	NaN	Cit

65535 rows × 14 columns

In [17]:
```
df["stop_duration"].mean()
```

Out[17]:
```
12.187420698181345
```

Groupby, Describe

v) compare the age distributions for each violation

In [20]:
```
df.groupby('violation').driver_age.describe()
```

Out[20]:

	count	mean	std	min	25%	50%	75%	max
violation								
Equipment	6507.0	31.682957	11.380671	16.0	23.0	28.0	39.0	81.0
Moving violation	11876.0	36.736443	13.258350	15.0	25.0	35.0	47.0	86.0
Other	3477.0	40.362381	12.754423	16.0	30.0	41.0	50.0	86.0
Registration/plates	2240.0	32.656696	11.150780	16.0	24.0	30.0	40.0	74.0
Seat belt	3.0	30.333333	10.214369	23.0	24.5	26.0	34.0	42.0
Speeding	37120.0	33.262581	12.615781	15.0	23.0	30.0	42.0	88.0

In [ ]:
```

```