In [1]:
```
# Imports
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# reading and printing all the columns
df = pd.read_csv('semifinal_police_20att.csv')
pd.set_option('display.max_columns', None)
#pd.set_option('display.max_rows', None)
df.info()
```

```
C:\Users\jules\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:344
4: DtypeWarning: Columns (1) have mixed types.Specify dtype option on import or
set low_memory=False.
  exec(code_obj, self.user_global_ns, self.user_ns)

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 872349 entries, 0 to 872348
Data columns (total 20 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   Year of Incident             872349 non-null  int64
 1   Watch                        872349 non-null  object
 2   Call (911) Problem           833166 non-null  object
 3   Type of Incident             872349 non-null  object
 4   Type  Location               871137 non-null  object
 5   Reporting Area               871305 non-null  float64
 6   Beat                         871966 non-null  float64
 7   Division                     871966 non-null  object
 8   Sector                       872099 non-null  float64
 9   Council District             870345 non-null  object
 10  Year1 of Occurrence          872349 non-null  int64
 11  Month1 of Occurence          872349 non-null  object
 12  Day1 of the Week             872349 non-null  object
 13  Time1 of Occurrence          872349 non-null  object
 14  Person Involvement Type      841122 non-null  object
 15  Victim Type                  833062 non-null  object
 16  Hate Crime Description       871082 non-null  object
 17  Drug Related Istevencident   833991 non-null  object
 18  Penal Code                   872349 non-null  object
 19  Zip Code                     868687 non-null  float64
dtypes: float64(4), int64(2), object(14)
memory usage: 133.1+ MB
```

Info on Numeric Data

In [2]:
```
df.describe()
```

Out[2]:

	Year of Incident	Reporting Area	Beat	Sector	Year1 of Occurrence	Zip Code
count	872349.000000	871305.000000	871966.000000	872099.000000	872349.000000	868687.000000
mean	2018.051145	3146.860872	415.544421	411.518211	2017.932530	75224.503813
std	25.639850	1827.654005	196.797775	196.950068	2.178059	163.115154
min	1014.000000	1001.000000	7.000000	0.000000	1974.000000	0.000000
25%	2016.000000	1247.000000	237.000000	230.000000	2016.000000	75214.000000
50%	2018.000000	3059.000000	421.000000	420.000000	2018.000000	75224.000000
75%	2020.000000	4323.000000	552.000000	550.000000	2020.000000	75236.000000
max	9999.000000	9611.000000	757.000000	750.000000	2021.000000	98004.000000

Handling Duplicates

In [3]:
```
df.duplicated().sum()
```

Out[3]:
```
23576
```

In [4]:
```
df = df.drop_duplicates()
```

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

Out[5]:
```
0
```

Year of Incidents

In [7]:
```
df['Year of Incident'].sort_values(ascending=True)
```

Out[7]:
```
532014    1014
48460     1211
49048     1429
37486     2010
43491     2013
          ...
45461     8633
42644     9578
147675    9898
368994    9999
324067    9999
Name: Year of Incident, Length: 848773, dtype: int64
```

In [13]:
```
yearIndex = df[(df['Year of Incident'] < 2000)].index
```

In [14]:
```
df[(df['Year of Incident'] > 2021)].index
```

Out[14]:
```
Int64Index([ 39648,  42169,  42644,  45461,  51401,  53483,  65918,  65937,
             74286,  76919,  76965,  78725,  83214, 102222, 119054, 119252,
            120632, 147675, 324067, 368994],
           dtype='int64')
```

In [37]:
```
yearIndex = df[(df['Year of Incident'] < 2000)].index
```

Out[37]:
```
Int64Index([], dtype='int64')
```

In [29]:
```
yearIndex = df[(df['Year of Incident'] > 2021)].index
```

In [18]:
```
df.shape
```

Out[18]:
```
(848773, 20)
```

In [31]:
```
df.drop(yearIndex , inplace=True)
```

In [39]:
```
df.shape
```

Out[39]:
```
(848750, 20)
```

In [38]:
```
df.describe()
```

Out[38]:

	Year of Incident	Reporting Area	Beat	Sector	Year1 of Occurrence	Zip Code
count	848750.000000	847728.000000	848378.000000	848509.000000	848750.000000	845244.000000
mean	2017.952769	3147.442722	415.894820	411.867146	2017.935815	75224.505456
std	2.167524	1826.621837	196.733498	196.887080	2.178771	165.331758
min	2010.000000	1001.000000	7.000000	0.000000	1974.000000	0.000000
25%	2016.000000	1247.000000	237.000000	230.000000	2016.000000	75214.000000
50%	2018.000000	3059.000000	421.000000	420.000000	2018.000000	75224.000000
75%	2020.000000	4323.000000	552.000000	550.000000	2020.000000	75236.000000
max	2021.000000	9611.000000	757.000000	750.000000	2021.000000	98004.000000

Number of missing attributes in an instance

In [71]:
```
df.isna().sum(1).sort_values(ascending = False).head(125)
```

Out[71]:
```
494735    8
227677    8
676527    7
118973    7
30324     7
         ..
81116     7
60619     7
716162    7
491742    7
80859     6
Length: 125, dtype: int64
```

Removing rows if missing values > 6

In [92]:
```
missingIndex = df[df.isna().sum(1) > 6].index
```

In [93]:
```
missingIndex
```

Out[93]:
```
Int64Index([ 26489,  30324,  36918,  42713,  43043,  43323,  43451,  45160,
             50260,  50295,
            ...
            790072, 795100, 800393, 806257, 809080, 830089, 842931, 845729,
            853922, 864825],
           dtype='int64', length=124)
```

In [94]:
```
df.drop(missingIndex , inplace=True)
```

In [95]:
```
df.shape
```

Out[95]:
```
(848626, 20)
```

Beat = 7

In [104]:
```
df[df['Beat'] == 7]
```

Out[104]:

	Year of Incident	Watch	Call (911) Problem	Type of Incident	Type Location	Reporting Area	Beat	Division	Sector
594462	2014	2	ODJ - OFF DUTY JOB	LOST PROPERTY (NO OFFENSE)	School/Daycare	NaN	7.0	South Central	70.0

In [108]:
```
beatIndex = df[df['Beat'] == 7].index
```

In [109]:
```
beatIndex
```

Out[109]:
```
Int64Index([594462], dtype='int64')
```

In [110]:
```
df.drop(beatIndex , inplace=True)
```

In [111]:
```
df.shape
```

Out[111]:
```
(848625, 20)
```

Sector = 0

In [112]:
```
df[df['Sector'] == 0]
```

					PROPERTY (NO OFFE...				
838587	2021	3		NaN	LOST PROPERTY (NO OFFENSE)	Other	NaN	NaN	N
844947	2015	1	58 - ROUTINE INVESTIGATION		LOST PROPERTY (NO OFFENSE)	Apartment Complex/Building	NaN	NaN	N
863602	2014	1	58 - ROUTINE INVESTIGATION		COMPUTER SECURITY BREACH	Single Family Residence - Occupied	NaN	NaN	N

In [114]:
```
sectorIndex = df[df['Sector'] == 0].index
```

In [115]:
```
sectorIndex
```

Out[115]:
```
Int64Index([   758,   2803,   2983,   3295,   3311,   3531,   5745,  59790,
             60044,  66970,
            ...
            786721, 798198, 800682, 813263, 821167, 821367, 831954, 838587,
            844947, 863602],
           dtype='int64', length=157)
```

In [116]:
```
df.drop(sectorIndex , inplace=True)
```

In [117]:
```
df.shape
```

Out[117]:
```
(848468, 20)
```

Zipcode = 0

In [120]:
```
df[df['Zip Code'] == 0]
```

Out[120]:

	Year of Incident	Watch	Call (911) Problem	Type of Incident	Type Location	Reporting Area	Beat	Division	Sector	C D
855362	2018	2	11V - BURG MOTOR VEH	BMV (OF AUTO ACCESSORY) (P.C. 30.04(A))	Parking Lot (All Others)	1027.0	642.0	NORTH CENTRAL	640.0	

In [122]:
```
zipcodeIndex = df[df['Zip Code'] == 0].index
```

In [123]:
```
df.drop(zipcodeIndex , inplace=True)
```

In [124]:
```
df.shape
```

Out[124]:
```
(848467, 20)
```

UNK to Unknown in Drug Related Istevencident

In [128]:
```
df[df['Drug Related Istevencident'] == 'UNK'].index
```

Out[128]:
```
Int64Index([     9,     10,     21,     23,     24,     31,     36,     55,
                62,     64,
            ...
            872222, 872242, 872250, 872254, 872276, 872283, 872290, 872304,
            872329, 872339],
           dtype='int64', length=81426)
```