Project: Investigate Crimes in Chicago

Table of Contents

Introduction

In this project, the dataset containing the information about crimes in chicago are explored. The primary objective of the project to make an attempt to explore the dataset and finds the relationship between different attributes. The dataset is called as “Chicago_Crimes_2005_to_2007” and contains around 1000000 tuples and 25 attributes including numeric and categorical. The dataset is downloaded from Kaggle. In this project, the crimes are analyzed over the period, crime type and location. The results and findings are presented in the report.

In [1]:
```
# Use this cell to set up import statements for all of the packages that you
#   plan to use.

# Remember to include a 'magic word' so that your visualizations are plotted
#   inline with the notebook. See this page for more:
#   http://ipython.readthedocs.io/en/stable/interactive/magics.html
# import libraries
import random
import math
import pandas as pd # to load csv file
import numpy as np
import matplotlib.pyplot as graph
from datetime import datetime
import seaborn as sns
```

Data Wrangling

This section loads the dataset using pandas library.

General Properties

In [2]:
```
# Load your data and print out a few lines. Perform operations to inspect data
#   types and look for instances of missing or possibly errant data.
# read dataset
dataFrame = pd.read_csv("Chicago_Crimes_2005_to_2007.csv",error_bad_lines=False)
dataFrame.head()
# format date attribute
dataFrame.index = pd.to_datetime(dataFrame.index)
dataFrame.Date = pd.to_datetime(dataFrame.Date, format = '%m/%d/%Y %I:%M:%S %p')
dataFrame.index = pd.DatetimeIndex(dataFrame.Date)
```

```
b'Skipping line 533719: expected 23 fields, saw 24\n'
```

In this phase, the data cleaning techniques are performed on the selected dataset. The data cleaning techniques are helps to remove empty cells from the dataset, these empty cells may cause problems in the results. The dataset contains the 'ID' attribute, this attribute is unique for each record so it is dropped from the dataset for further analysis. The tuples contains the value na or inf are delete from the dataset.

Data Cleaning (remove rows contains 'na' or 'inf')

In [3]:
```
# After discussing the structure of the data and any problems that need to be
#   cleaned, perform those cleaning steps in the second part of this section.
# remove all na/empty rows
cleandf = dataFrame.replace([np.inf, -np.inf], np.nan).dropna(axis=0)
cleandf.head()
```

Out[3]:

Date	Unnamed: 0	ID	Case Number	Date	Block	IUCR	Primary Type	Desc
2006-04-02 13:00:00	0	4673626	HM274058	2006-04-02 13:00:00	055XX N MANGO AVE	2825	OTHER OFFENSE	HARASSM TELE
2006-02-26 13:40:48	1	4673627	HM202199	2006-02-26 13:40:48	065XX S RHODES AVE	2017	NARCOTICS	MANU/DELIVER:
2006-01-08 23:16:00	2	4673628	HM113861	2006-01-08 23:16:00	013XX E 69TH ST	051A	ASSAULT	AGGRA HA
2006-04-05 18:45:00	4	4673629	HM274049	2006-04-05 18:45:00	061XX W NEWPORT AVE	0460	BATTERY	
2006-02-17 21:03:14	5	4673630	HM187120	2006-02-17 21:03:14	037XX W 60TH ST	1811	NARCOTICS	POSS: CA 30GMS O

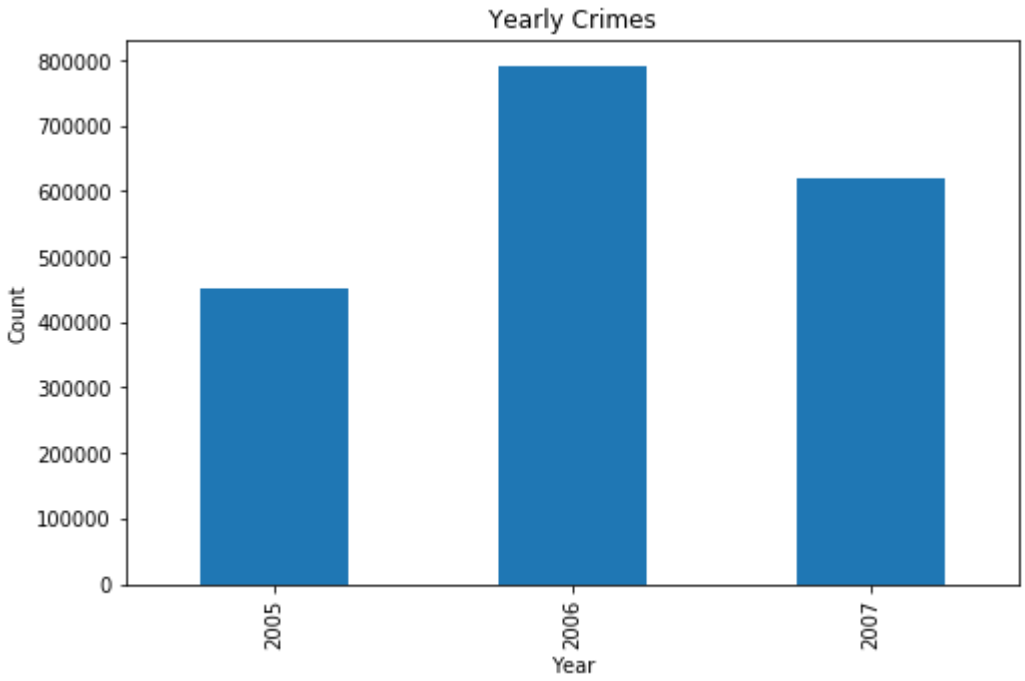
5 rows × 23 columns

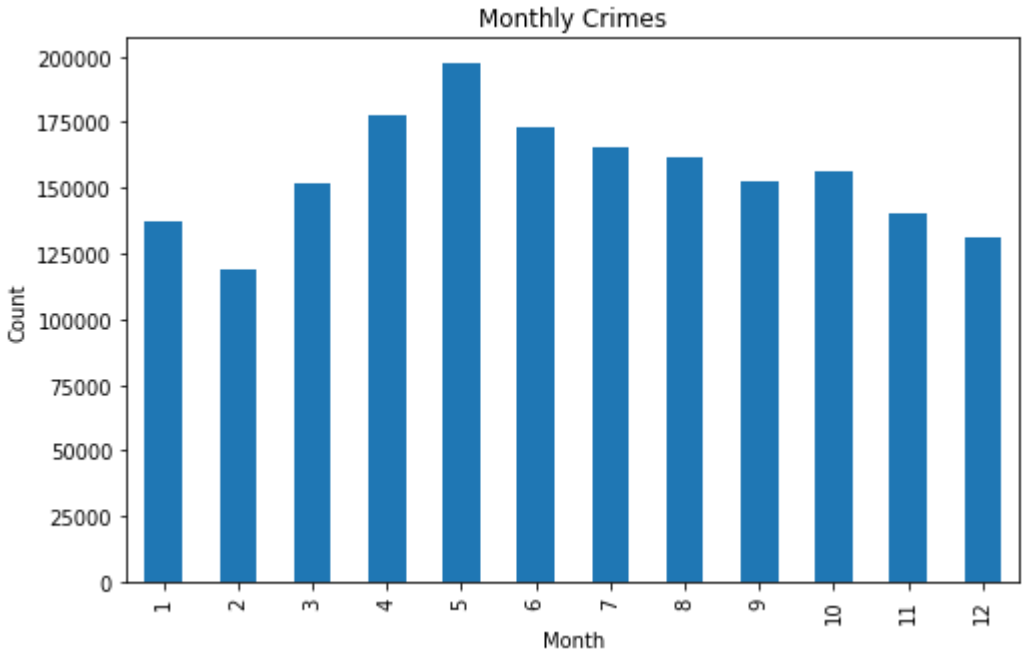
Exploratory Data Analysis

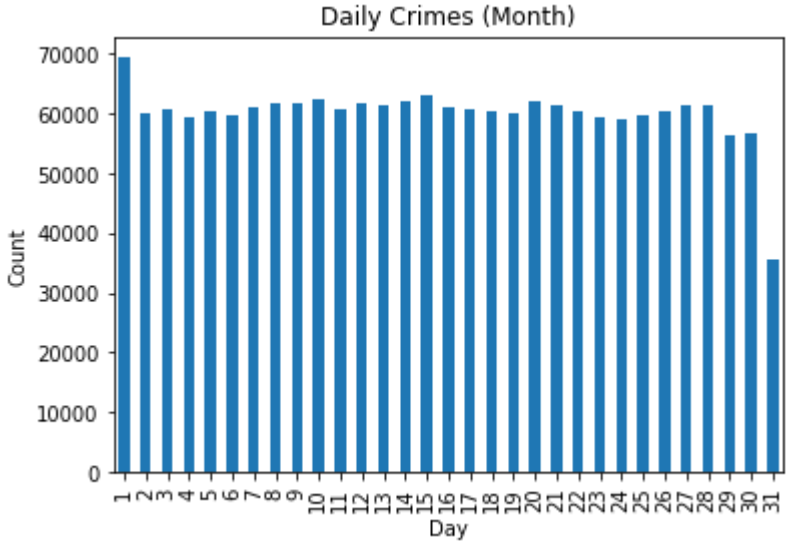
The objective of the project is to analyze real time dataset. To selected dataset contains the 25 attributes: Hypothesis: To know the relationship between Crimes and period To know the relationship between Crimes and Type and cluster the crimes by Type To know the relationship between Crimes and Location and cluster the crimes by Location

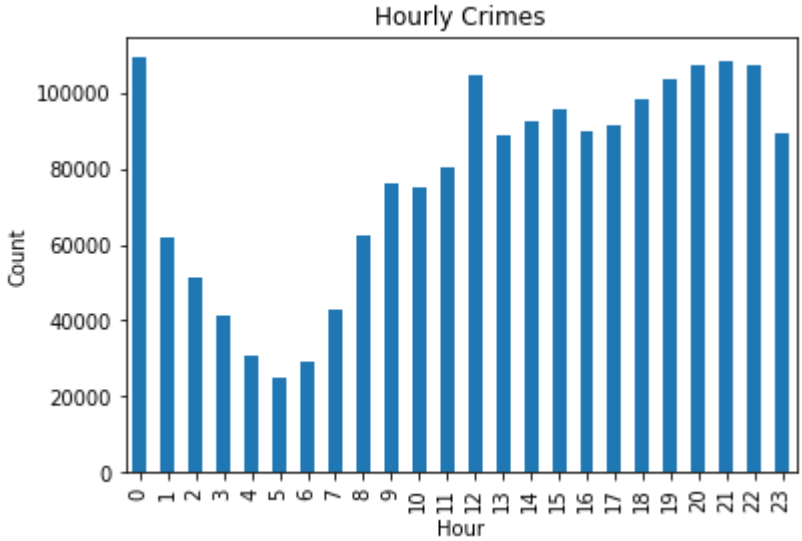
Research Question 1 (Crimve over the period)

In [11]:
```
# Use this, and more code cells, to explore your data. Don't forget to add
#   Markdown cells to document your observations and findings.
# plot statistics to analyze dataset
graph.figure(figsize = (8,5))
cleandf.groupby([cleandf.index.year]).size().plot.bar()
graph.title('Yearly Crimes')
graph.xlabel('Year')
graph.ylabel('Count')
graph.show()
graph.figure(figsize = (8,5))

cleandf.groupby([cleandf.index.month]).size().plot.bar()
graph.title('Monthly Crimes')
graph.xlabel('Month')
graph.ylabel('Count')
graph.show()

cleandf.groupby([cleandf.index.day]).size().plot.bar()
graph.title('Daily Crimes (Month)')
graph.xlabel('Day')
graph.ylabel('Count')
graph.show()

cleandf.groupby([cleandf.index.hour]).size().plot.bar()
graph.title('Hourly Crimes')
graph.xlabel('Hour')
graph.ylabel('Count')
graph.show()
```




Monthly Crimes
Count
Month
0
25000
50000
75000
100000
125000
150000
175000
200000
1
2
3
4
5
6
7
8
9
10
11
12


Daily Crimes (Month)
Count
Day
0
10000
20000
30000
40000
50000
60000
70000


Hourly Crimes
Count
Hour
0
20000
40000
60000
80000
100000

In [12]:
```
arrestedbyYears = pd.DataFrame(pd.DataFrame(cleandf ,columns=['Year','Arrest']).
                               groupby(['Year','Arrest']).size().sort_values(asc
print('Arrest/Not Arrest over the Period')
print(arrestedbyYears)
```

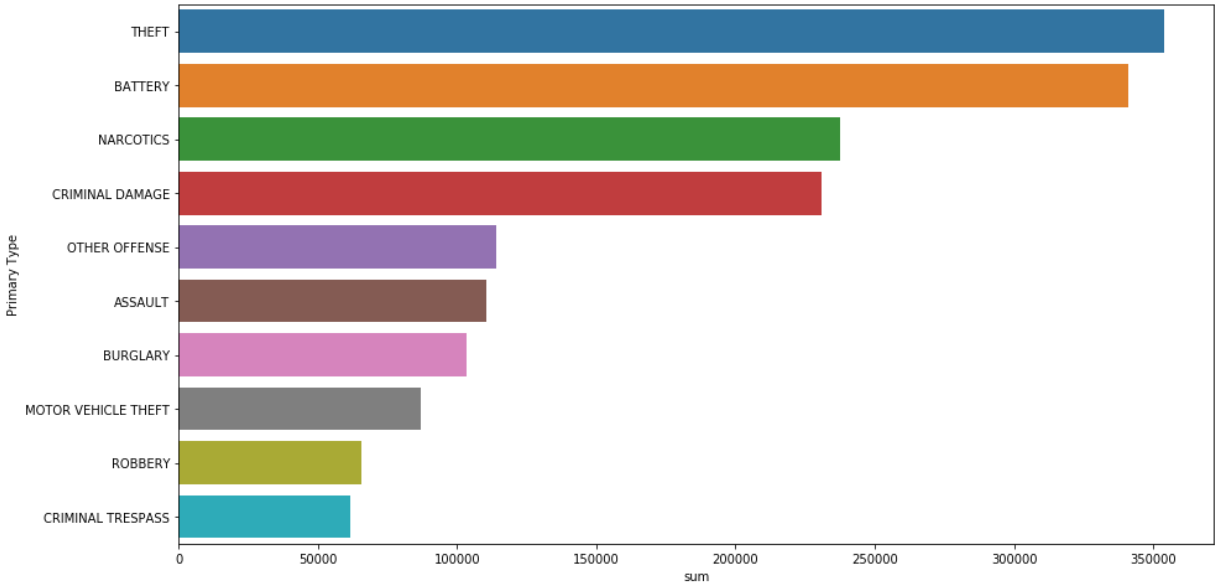
```
Arrest/Not Arrest over the Period
              Total
Year Arrest
2006 False   548988
2007 False   430372
2005 False   310289
2006 True    242970
2007 True    189932
2005 True    140281
```

In [13]:
```
topCrimes = pd.DataFrame( pd.DataFrame(cleandf ,columns=['Primary Type']).
                               groupby(['Primary Type']).size().sort_values(asce
print("Top 10 Crimes")
print(topCrimes.head(10))
```

```
Top 10 Crimes
          Primary Type   Total
0                THEFT  353971
1              BATTERY  340852
2            NARCOTICS  237537
3      CRIMINAL DAMAGE  230857
4        OTHER OFFENSE  113943
5              ASSAULT  110526
6             BURGLARY  103269
7  MOTOR VEHICLE THEFT   86792
8              ROBBERY   65795
9    CRIMINAL TRESPASS   61624
```

Research Question 2 (Group crimes by type)

In [14]:
```
# Continue to explore the data to address your additional research
#   questions. Add more headers as needed if you have more questions to
#   investigate.
# count crime count for each type
crimesbyType = pd.DataFrame(cleandf.groupby('Primary Type').size().sort_values(a
crimesbyType.head()

f, ax = graph.subplots(figsize=(15, 8))
sns.barplot(sns.color_palette("hls", 10))
sns.barplot(y="Primary Type", x="sum", data=crimesbyType.iloc[:10, :])
graph.show()
```



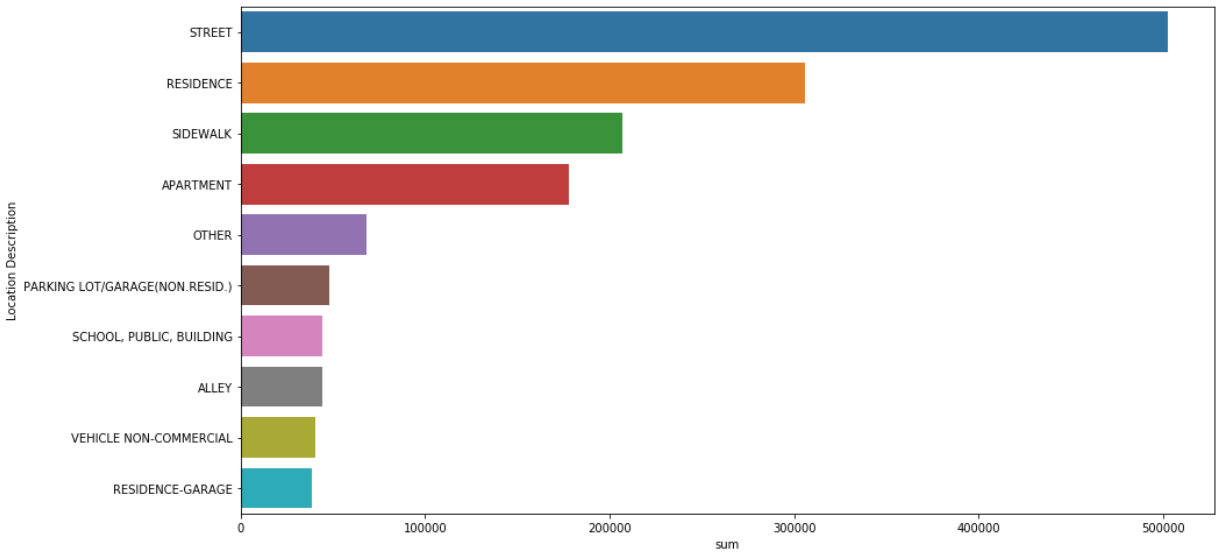
In [15]:
```
# group crime type by arrest
groupbyArrested = pd.DataFrame( pd.DataFrame(cleandf,columns=['Primary Type','Ar
print(groupbyArrested)
```

	Primary Type	Arrest	counts
0	THEFT	False	310987
1	BATTERY	False	259532
2	NARCOTICS	True	234510
3	CRIMINAL DAMAGE	False	213775
4	BURGLARY	False	96931
5	OTHER OFFENSE	False	96189
6	ASSAULT	False	83364
7	BATTERY	True	81320
8	MOTOR VEHICLE THEFT	False	78418
9	ROBBERY	False	58805
10	CRIMINAL TRESPASS	True	49038
11	DECEPTIVE PRACTICE	False	43929
12	THEFT	True	42984
13	PROSTITUTION	True	27532
14	ASSAULT	True	27162
15	OTHER OFFENSE	True	17754
16	CRIMINAL DAMAGE	True	17082
17	WEAPONS VIOLATION	True	12750
18	CRIMINAL TRESPASS	False	12586
19	DECEPTIVE PRACTICE	True	12173
20	OFFENSE INVOLVING CHILDREN	False	8505
21	PUBLIC PEACE VIOLATION	True	8475
22	MOTOR VEHICLE THEFT	True	8374
23	ROBBERY	True	6990
24	BURGLARY	True	6338
25	GAMBLING	True	5512
26	CRIM SEXUAL ASSAULT	False	4991
27	LIQUOR LAW VIOLATION	True	4727
28	PUBLIC PEACE VIOLATION	False	4544
29	SEX OFFENSE	False	4187
30	WEAPONS VIOLATION	False	3196
31	NARCOTICS	False	3027
32	OFFENSE INVOLVING CHILDREN	True	2686
33	INTERFERENCE WITH PUBLIC OFFICER	True	2661
34	ARSON	False	2532
35	SEX OFFENSE	True	2327
36	KIDNAPPING	False	1275
37	CRIM SEXUAL ASSAULT	True	951
38	INTIMIDATION	False	927
39	HOMICIDE	True	773
40	STALKING	False	680
41	HOMICIDE	False	609
42	ARSON	True	476
43	INTERFERENCE WITH PUBLIC OFFICER	False	252
44	PROSTITUTION	False	240
45	INTIMIDATION	True	191
46	KIDNAPPING	True	186
47	STALKING	True	113
48	GAMBLING	False	66

49	LIQUOR LAW VIOLATION	False	66
50	OBSCENITY	True	56
51	OTHER NARCOTIC VIOLATION	True	23
52	PUBLIC INDECENCY	True	18
53	OTHER NARCOTIC VIOLATION	False	14
54	RITUALISM	False	13
55	OBSCENITY	False	9
56	RITUALISM	True	1

Research Question 3 (Group crimes by Location)

In [16]:
```
# Continue to explore the data to address your additional research
#   questions. Add more headers as needed if you have more questions to
#   investigate.
# count crime count for each type
crimesbyLoc = pd.DataFrame(cleandf.groupby('Location Description').size().sort_va
crimesbyLoc.head()

f, ax = graph.subplots(figsize=(15, 8))
sns.barplot(sns.color_palette("hls", 10))
sns.barplot(x="sum",y="Location Description", data=crimesbyLoc.iloc[:10, :])
graph.show()
```



In [17]:
```
# group crime type by location
groupbyLocation = pd.DataFrame(pd.DataFrame(cleandf,columns=['Location Descriptio
print(groupbyLocation)
```

	Location Description	counts
0	STREET	502409
1	RESIDENCE	305681
2	SIDEWALK	206782
3	APARTMENT	178013
4	OTHER	68231
5	PARKING LOT/GARAGE(NON.RESID.)	48026
6	SCHOOL, PUBLIC, BUILDING	44450
7	ALLEY	44090
8	VEHICLE NON-COMMERCIAL	40264
9	RESIDENCE-GARAGE	38884
10	RESIDENCE PORCH/HALLWAY	32439
11	SMALL RETAIL STORE	29866
12	RESTAURANT	25626
13	GROCERY FOOD STORE	23183
14	GAS STATION	21484
15	CHA PARKING LOT/GROUNDS	19255
16	DEPARTMENT STORE	19223
17	PARK PROPERTY	14452
18	COMMERCIAL / BUSINESS OFFICE	14396
19	CHA HALLWAY/STAIRWELL/ELEVATOR	11214
20	CTA PLATFORM	10874
21	CHA APARTMENT	9342
22	DRUG STORE	8888
23	BANK	8376
24	SCHOOL, PUBLIC, GROUNDS	8334
25	BAR OR TAVERN	8251
26	AIRPORT/AIRCRAFT	7658
27	HOTEL/MOTEL	6973
28	TAVERN/LIQUOR STORE	6608
29	DRIVEWAY - RESIDENTIAL	6448
..	...	...
94	GARAGE	7
95	GAS STATION DRIVE/PROP.	7
96	BASEMENT	6
97	VESTIBULE	6
98	CHA GROUNDS	5
99	GANGWAY	4
100	RAILROAD PROPERTY	4
101	TAVERN	4
102	CLUB	3
103	STAIRWELL	3
104	BARBER SHOP/BEAUTY SALON	2
105	CHA LOBBY	2
106	HOTEL	2
107	SCHOOL YARD	2
108	VEHICLE - DELIVERY TRUCK	1
109	WOODED AREA	1
110	BANQUET HALL	1
111	LIVERY AUTO	1

```
112                     COACH HOUSE       1
113                           TRUCK       1
114                            LAKE       1
115                         TRAILER       1
116                  CHA STAIRWELL       1
117                           MOTEL       1
118                      RIVER BANK       1
119                           RIVER       1
120                        DRIVEWAY       1
121                        DUMPSTER       1
122              GARAGE/AUTO REPAIR       1
123                            YMCA       1

[124 rows x 2 columns]
```

Conclusions

This report presents the detailed analysis for the selected dataset. The report depcited the analysis of crimes in terms of period, crime type and location. From the resuls, it is concluded that the ratio of crimes in 2006 was high as compared to other years and most of the crimes were regarding to Theft. The count of Stree crimes are high while the crimes in residences are second number and the crimes in residence carage are the lowest.