CRIME DATA ANALYSIS IN CHICAGO (2020 to TILL DATE)

SNEHA RAJULAPALLY

A20457266

FINAL PROJECT

ITMD 513

srajulapally@hawk.iit.edu

Table of Contents

1. Abstract

In this project we are going to conduct crime analysis in city of Chicago starting in year 2020 to till date. The city's overall crime rate is substantially higher than the US average crime rate. Keeping this in mind, we have fetched dataset to analyze the trend in latest crime events that are occurring in the city to understand the crime patterns.

2. Introduction

Crime has been a prevalent anti-social trait in human society. Surging crime rates have been major problem. In order to record the crime in Chicago, the Chicago police department developed a tool to assist city residents in combating crime and to maintain transparency. Understanding crime patterns and criminal behaviors would largely be helpful for police force and for public. We are using the dataset provided by Chicago police department that reflects the reported incidents of crime. Data is extracted from Citizen Law Enforcement Analysis and Reporting System. We are using some data science technologies with combination of python libraries to get the desired results.

3. Project Overview

1. Focus

Crimes of Chicago Dataset has been extracted from the below URL and filtered out to get data of 2020 to till date and converted into CSV file.

https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2/data

2. Scope

- GUI login for user to login to the application
- User name and password are validated using SQLite3 database
- If new user, allowing to signup and create new account and update SQLite3 database

- After successful login, user to select from the radio button options to view the data/graph
- Close and exit options to logout or close the application

3. Files

- GUI_finaproject.py -> runs as main file to open the application
- DB_finalproject.py -> Load and user credential validations using SQLite3 Database
- Dataplots_finalproject.py -> get the data and display the plots, charts, graphs based on user selection
- Jupyter_finalproject.py ->works same as above
- Userdsdata.csv -> SQLite3 DB will validate login credentials using this file
- Crimes_-_2020.csv -> read the data from the csv file and display the results based on user selection in GUI.
- Finalproject.db -> DB connection is created

4. Libraries

- Tkinter
- Sqlite3
- Hashlib
- Pandas
- Matplotlib
- Numpy
- Seaborn
- Datetime
- Csv

4. Functionalities

1. Login Screen

1. SQLite3 DB

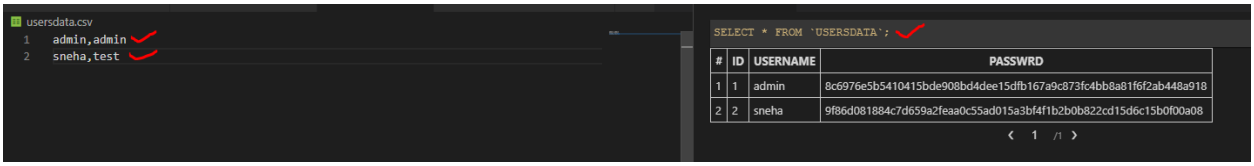
a. Finalproject.db is created with USERSDATA table.

USERSDATA table is populated with records in usersdata.csv file
Passwords are hashed in DB table.

User Credentials:

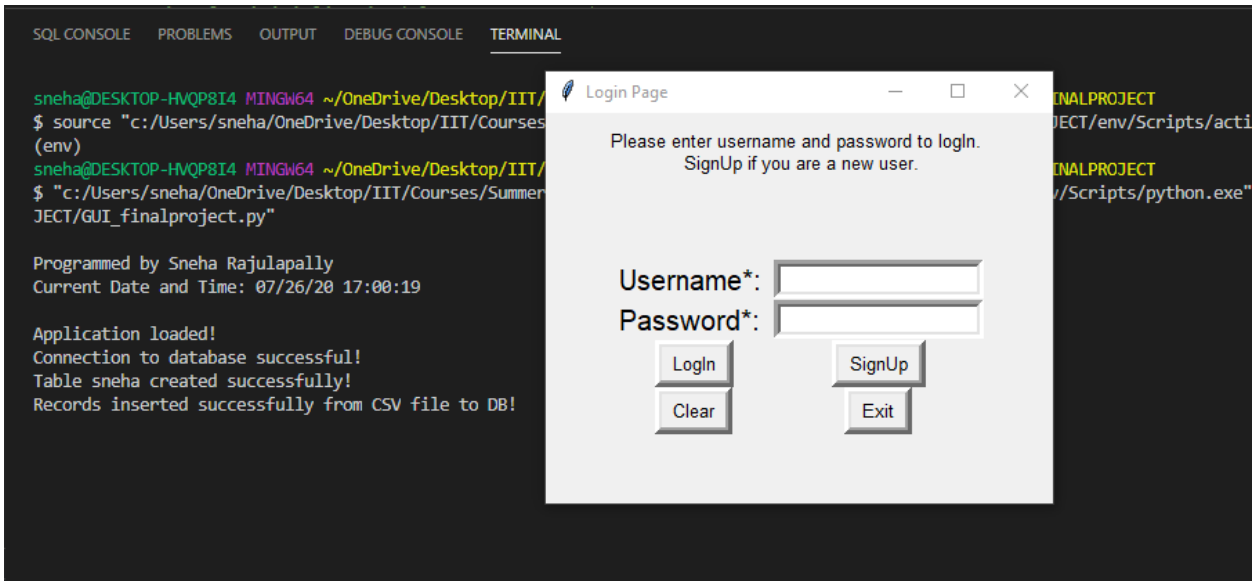
Username : admin
Password: admin

Username: sneha
Password: test



2. GUI

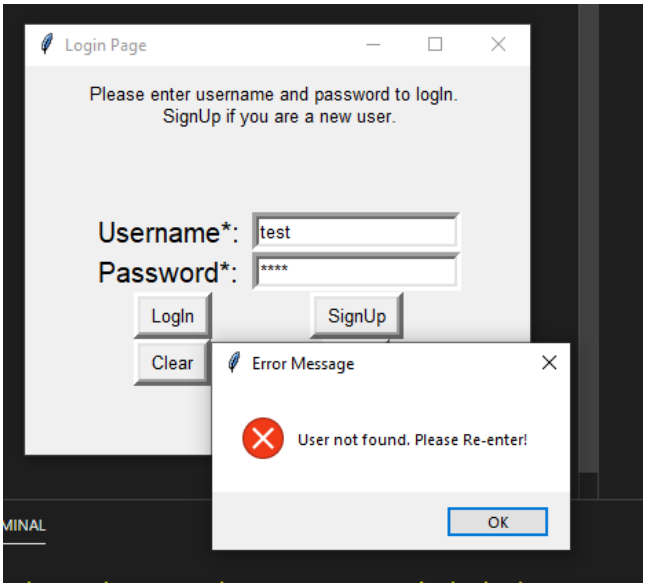
b. Run GUI_finalproject.py file to load the application.



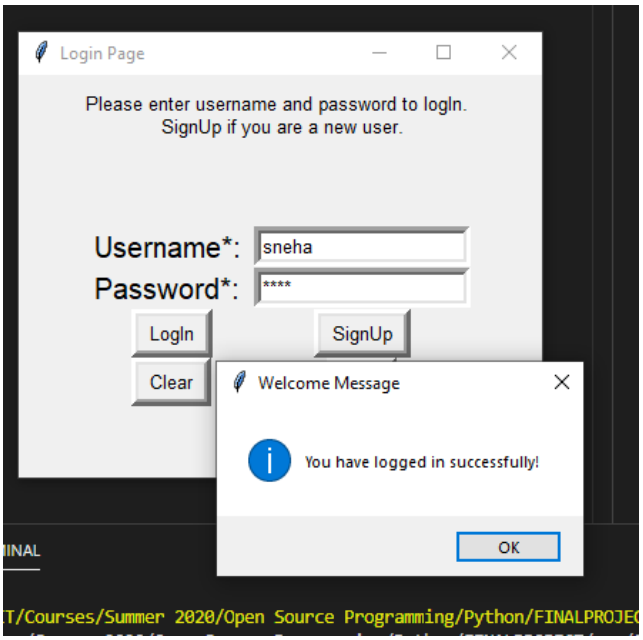
3. User credential validations

c. Enter Invalid credentials that are not present in CSV file or DB table.

- Message box with error appears, saying user not found. Please re-enter.



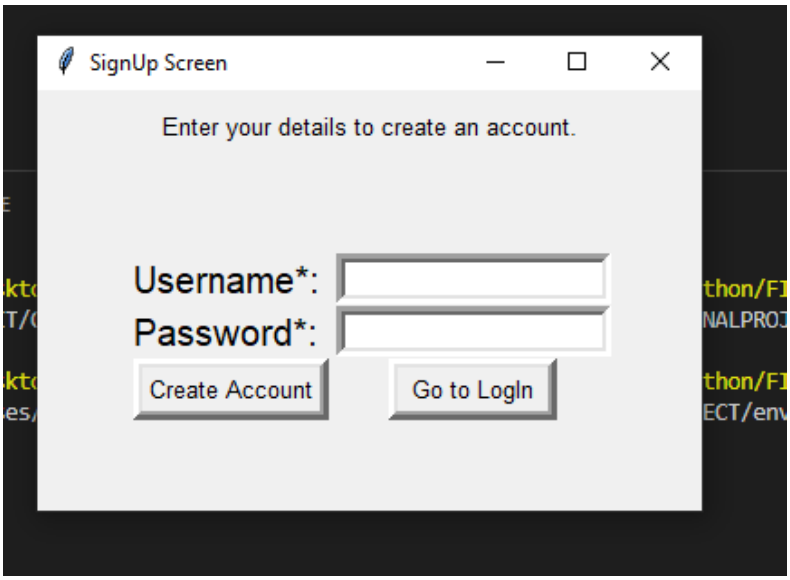
d. When correct credentials are entered, information with message box appears that you have successfully logged in.



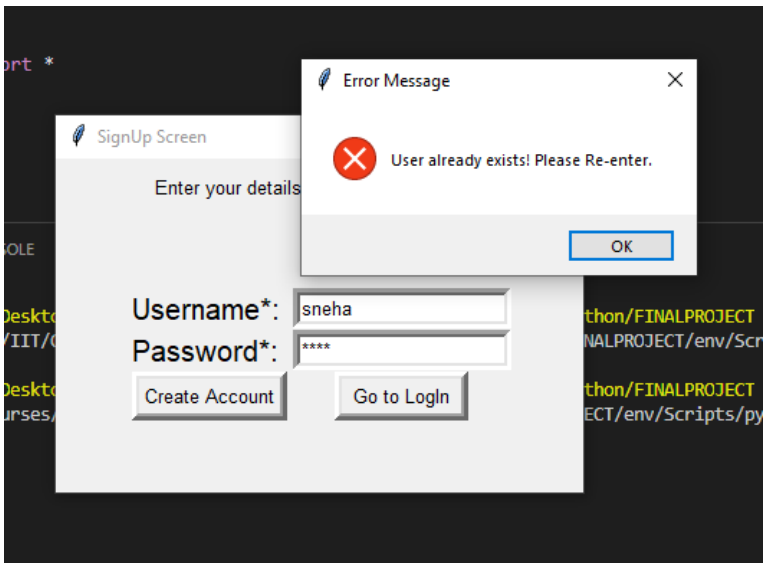
2. SignUp Screen

1. GUI

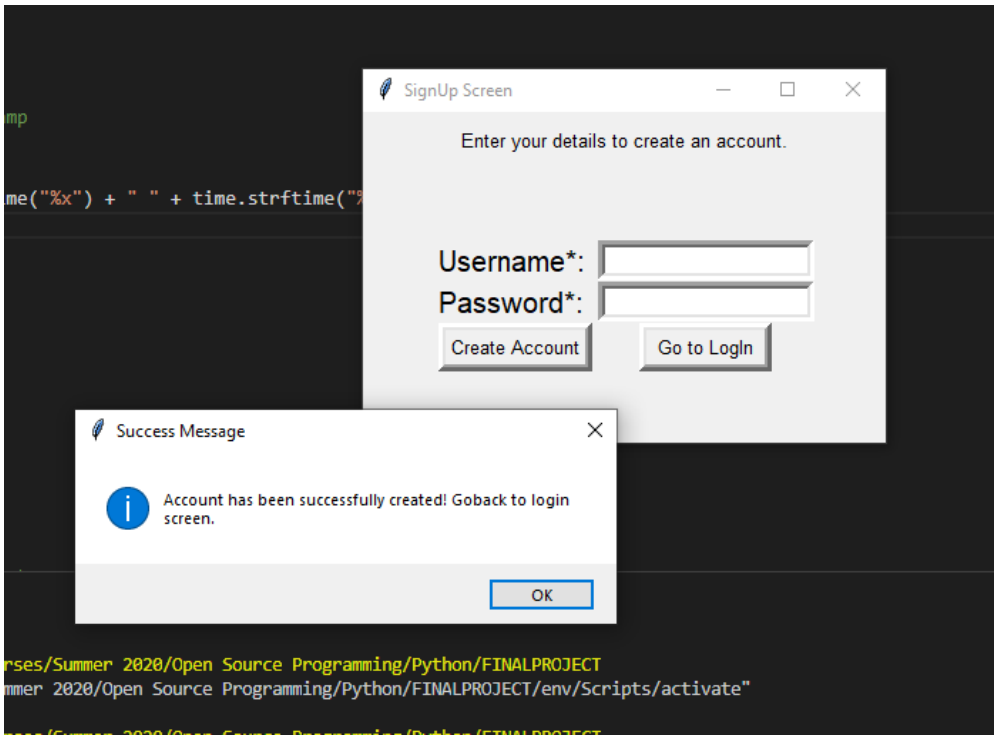
a. This is signup screen.
   - Create button allows to create new account after validation.
   - Go to login button takes back to login screen



b. Throw an error when you try to create existing user.
   - Message box appears that the user already exists.

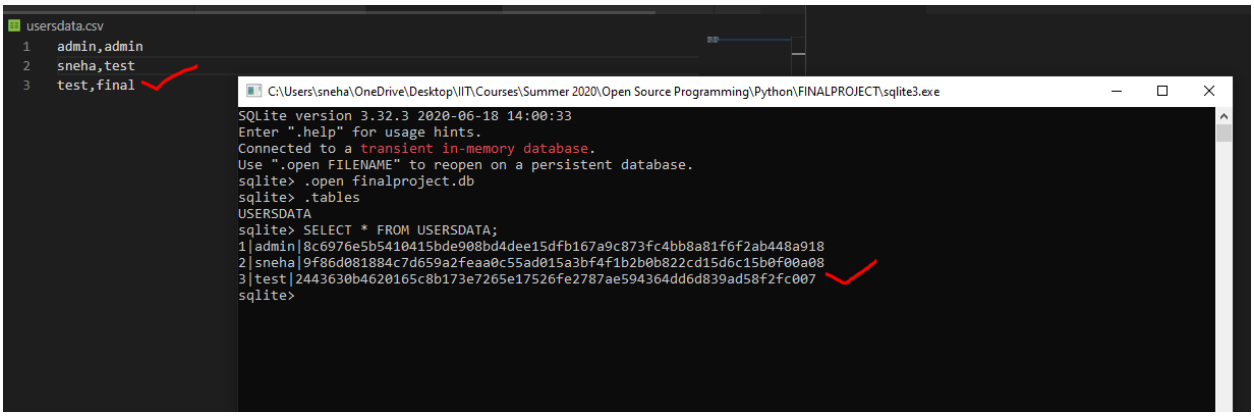


c. Throw a success message, when you add new users
   - Message box appears that the account has been creates successfully.



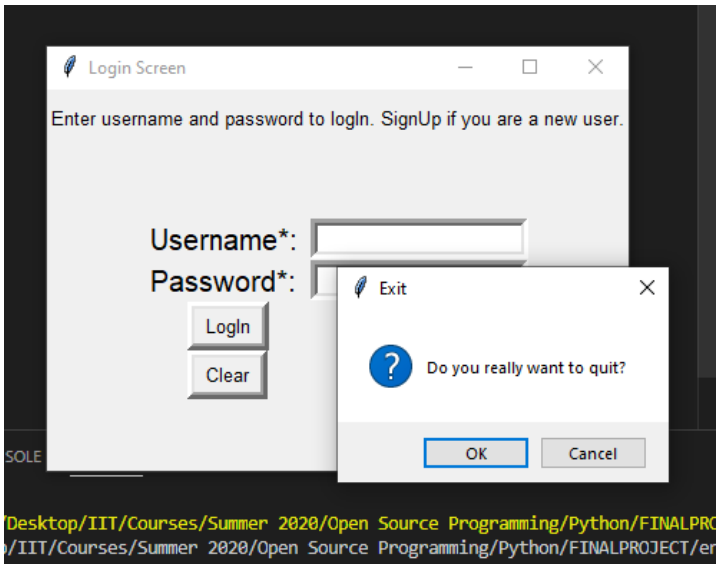
2. SQLite3 DB

a. SQLite 3 DB has been updates with new record and CSV file has been updated.



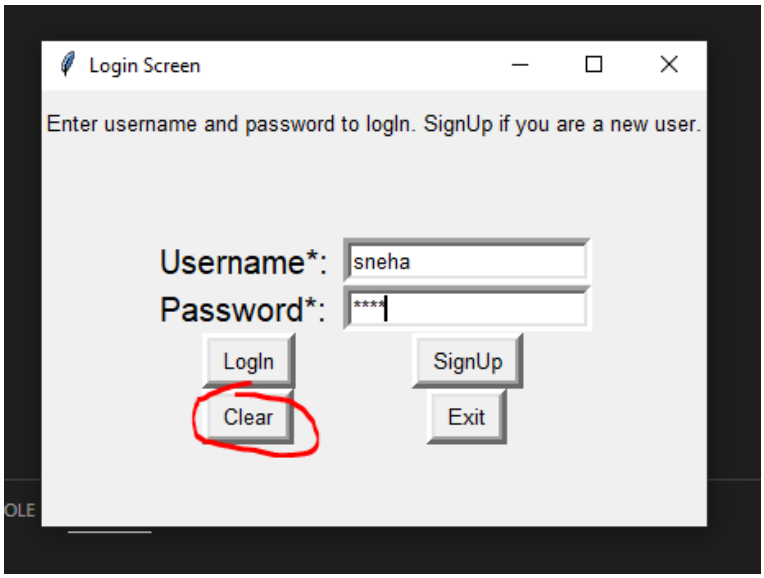
3. Exit Button

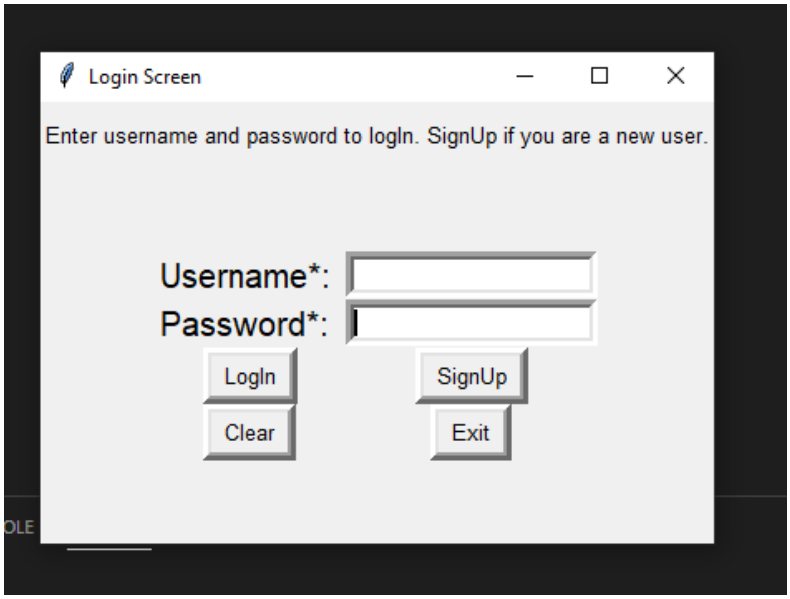
a. When exit button is clicked, message box appears if you want to quit application.



4. Clear Button

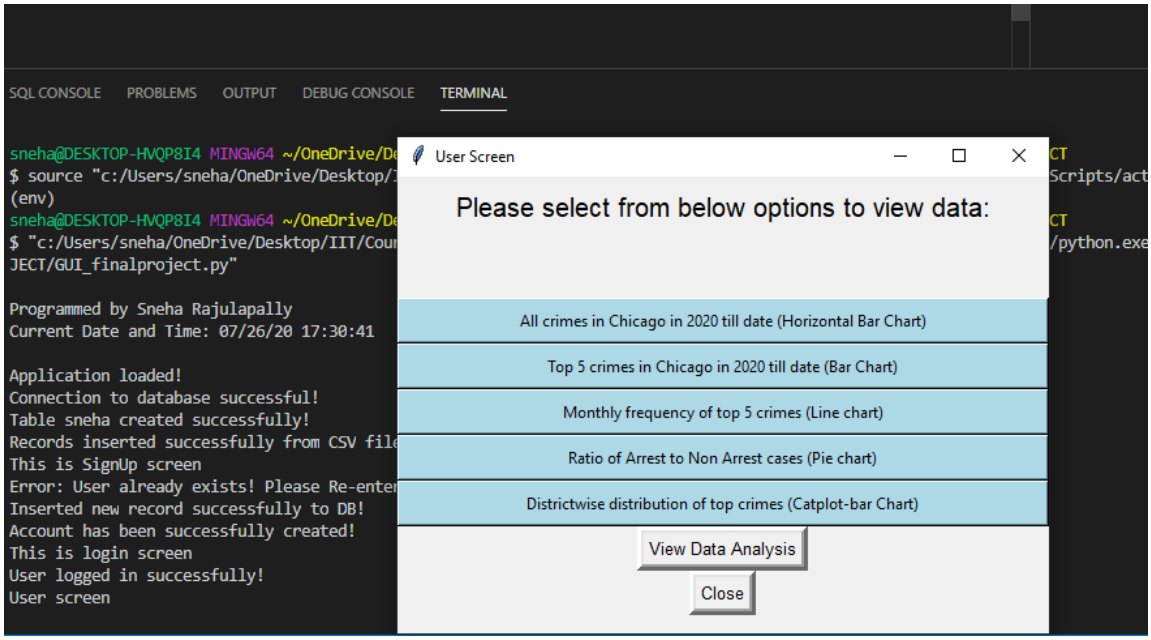
a. When clear button is clicked, the input texts will be cleared.



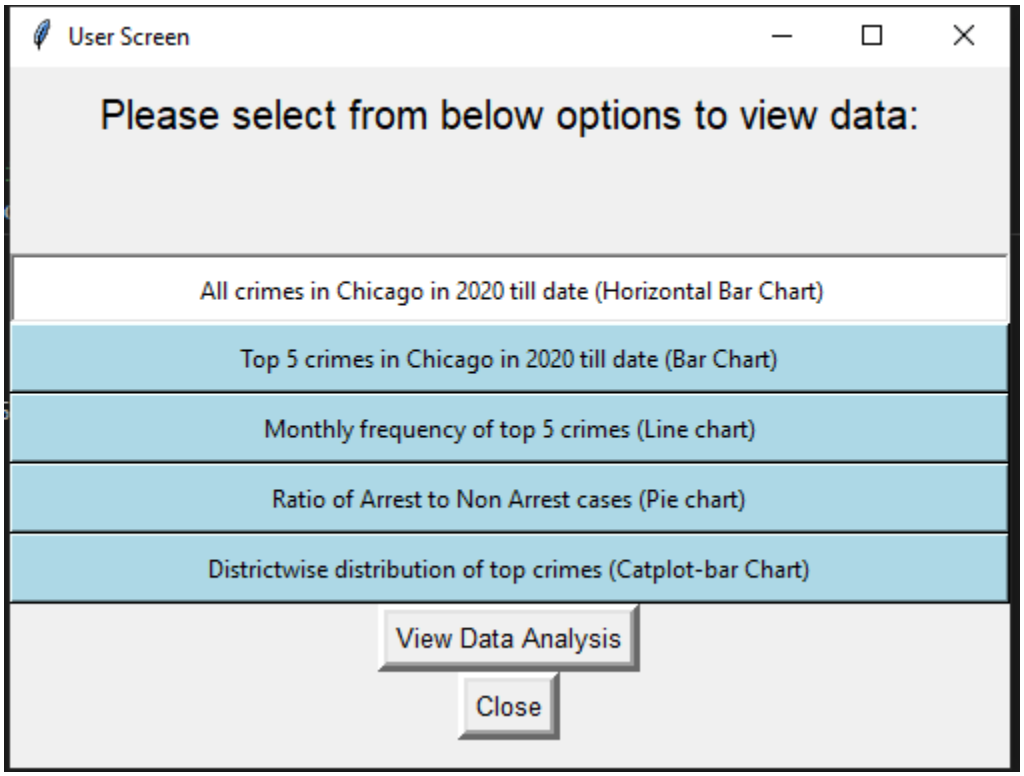


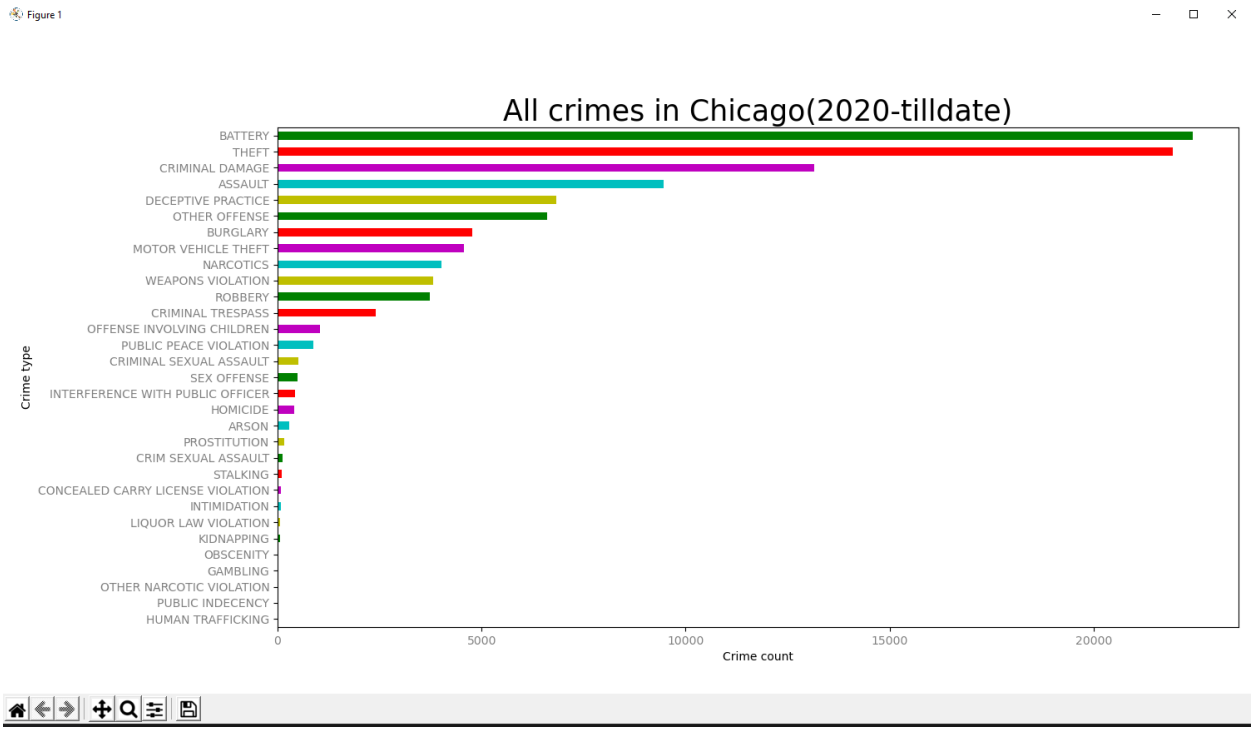
5. User Screen

a. After successful login, users screen is appeared.
   - View data analysis button is clicked after selecting any button from the 5 choices to view data.
   - Close button, exists you from the application



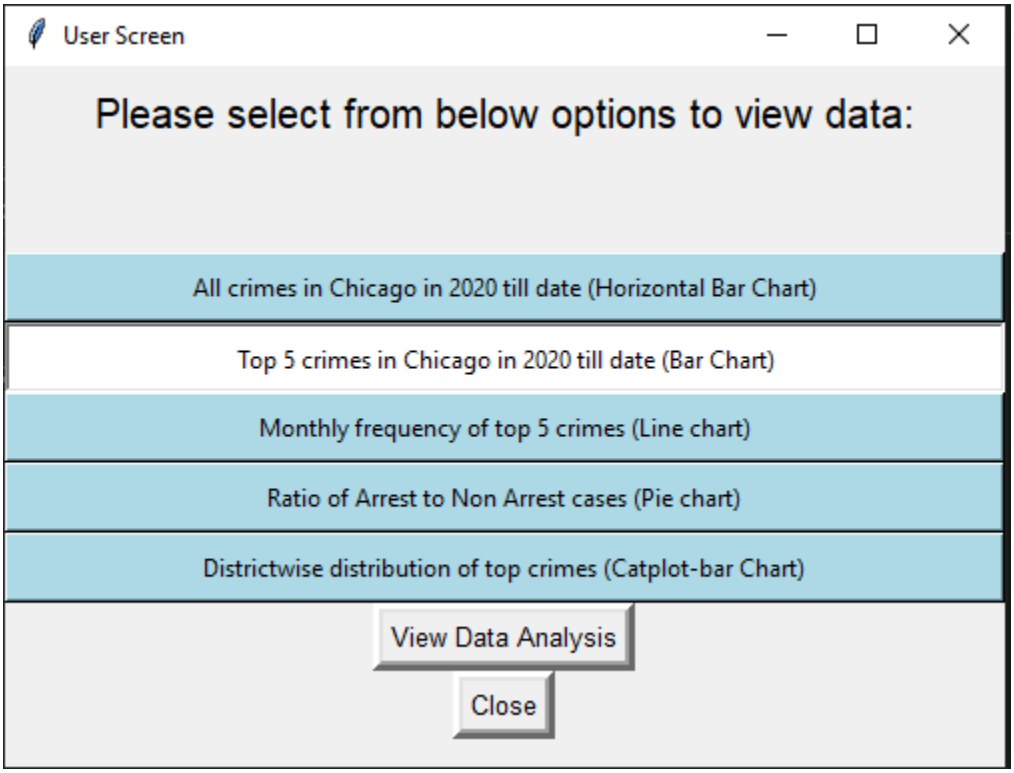
1. Radio Button 1: All crimes in Chicago in 2020 till date (Horizontal Bar Chart)

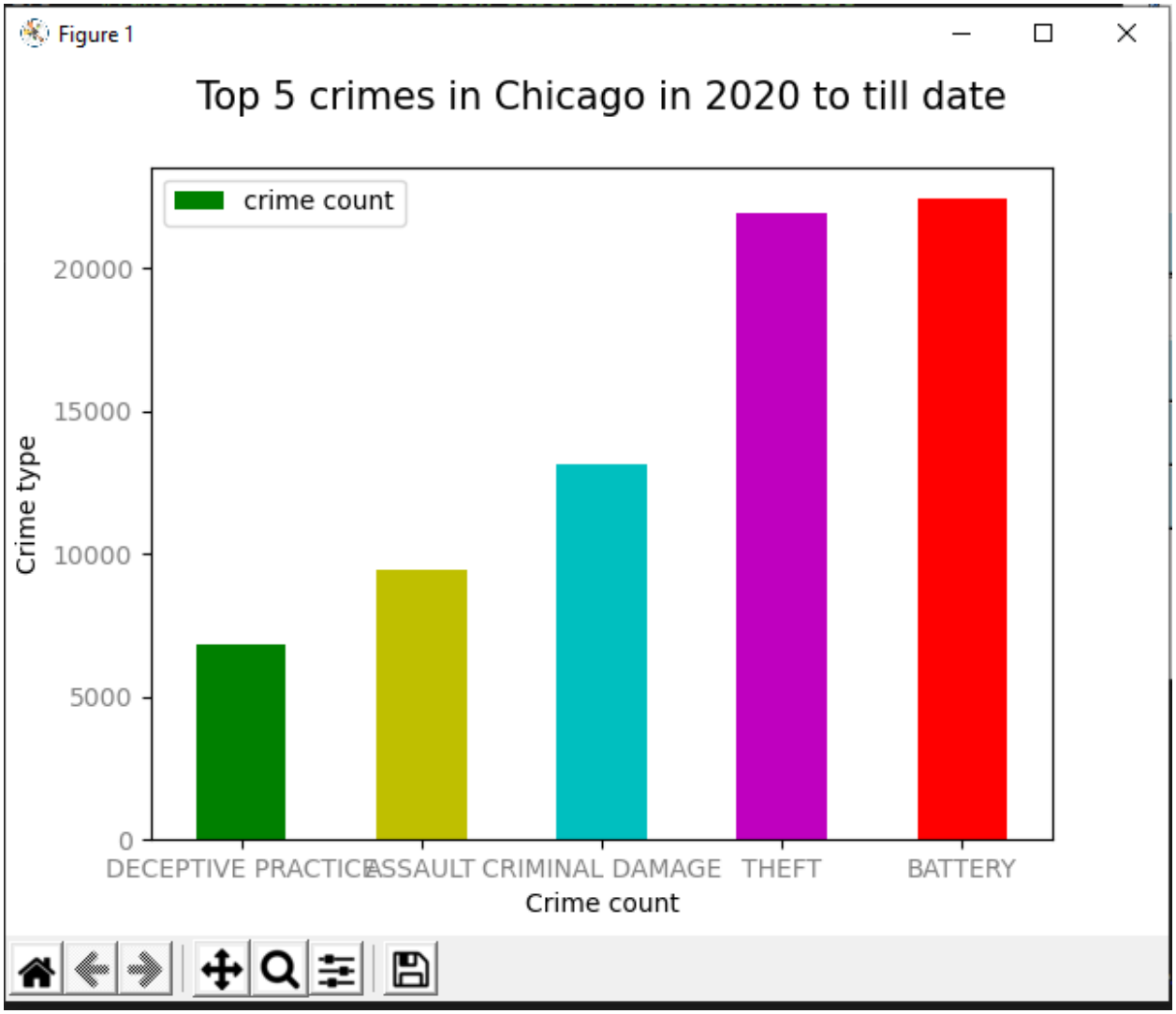




```
User logged in successfully!
User screen
Your selection is 1
Chicago crime Data for year 2020 has been successfully fetched from CSV file.
Total number of crime types in chicago in year 2020 to till date:  31
Crimetypes:
 ['BURGLARY' 'MOTOR VEHICLE THEFT' 'THEFT' 'OTHER OFFENSE'
 'CRIMINAL DAMAGE' 'ROBBERY' 'CRIMINAL SEXUAL ASSAULT'
 'DECEPTIVE PRACTICE' 'WEAPONS VIOLATION' 'OFFENSE INVOLVING CHILDREN'
 'HOMICIDE' 'ASSAULT' 'BATTERY' 'NARCOTICS'
 'INTERFERENCE WITH PUBLIC OFFICER' 'SEX OFFENSE' 'INTIMIDATION'
 'STALKING' 'CRIMINAL TRESPASS' 'PUBLIC PEACE VIOLATION' 'ARSON'
 'GAMBLING' 'LIQUOR LAW VIOLATION' 'OBSCENITY'
 'CONCEALED CARRY LICENSE VIOLATION' 'KIDNAPPING' 'PROSTITUTION'
 'HUMAN TRAFFICKING' 'PUBLIC INDECENCY' 'OTHER NARCOTIC VIOLATION'
 'CRIM SEXUAL ASSAULT']

Displaying data of all crimes commited in Chicago in 2020 till date using horizontal bar graph.
```

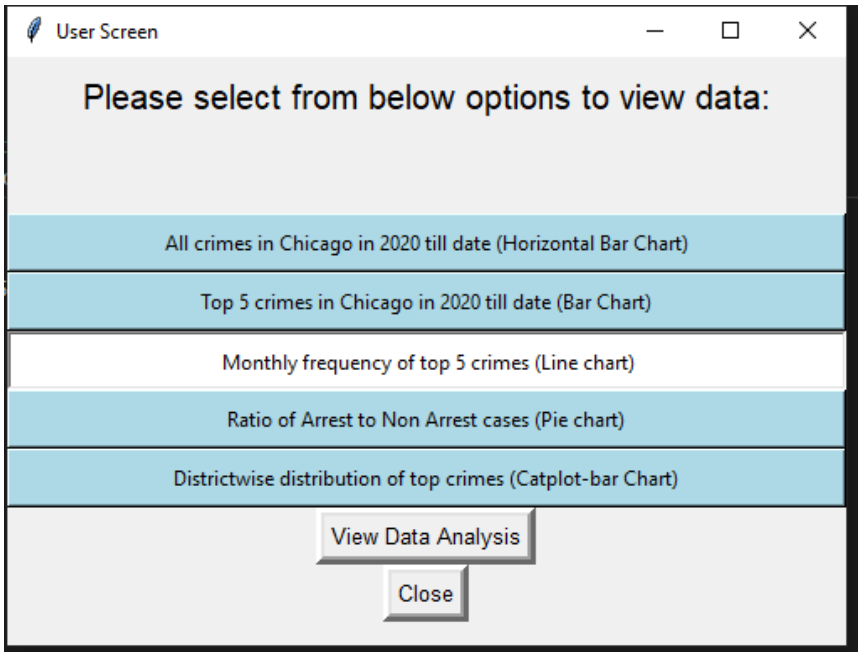
2. Radio Button 2: Top 5 crimes in Chicago in 2020 till date (Bar Chart)

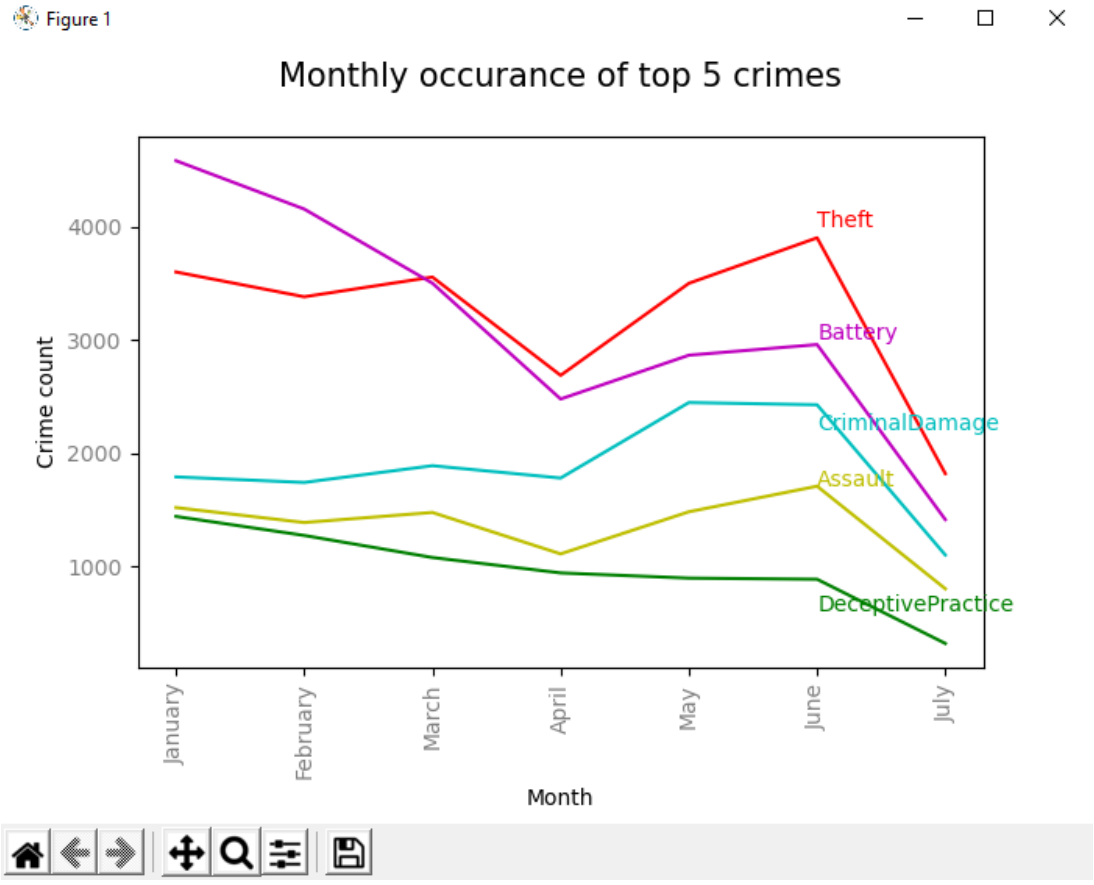





```
Your selection is 2
Chicago crime Data for year 2020 has been successfully fetched from CSV file.
Top 5 crimes and counts:
          Primary Type     ID
9   DECEPTIVE PRACTICE   6821
1              ASSAULT   9470
6      CRIMINAL DAMAGE  13160
29              THEFT  21941
2              BATTERY  22430

Displaying data of top5 crimes commited in Chicago in 2020 till date using bar graph.
```

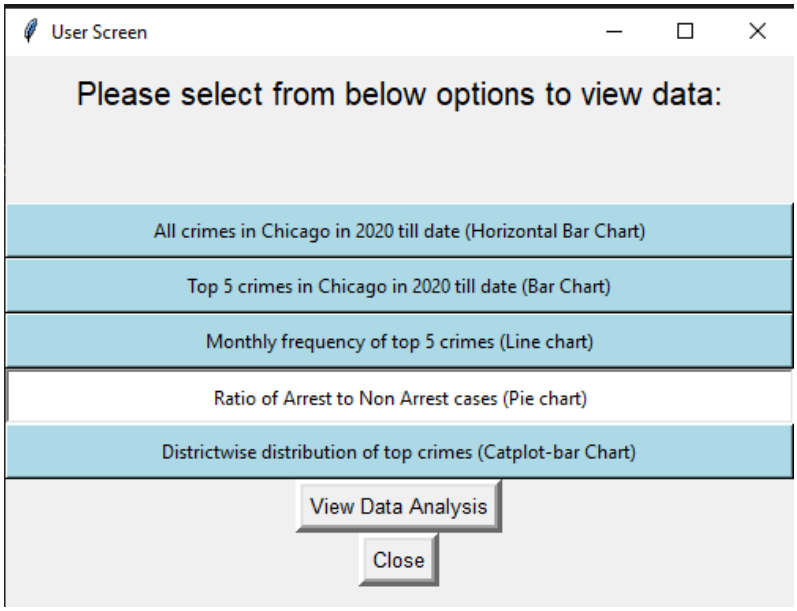
3. Radio Button 3: Monthly frequency of top 5 crimes (Line Chart)

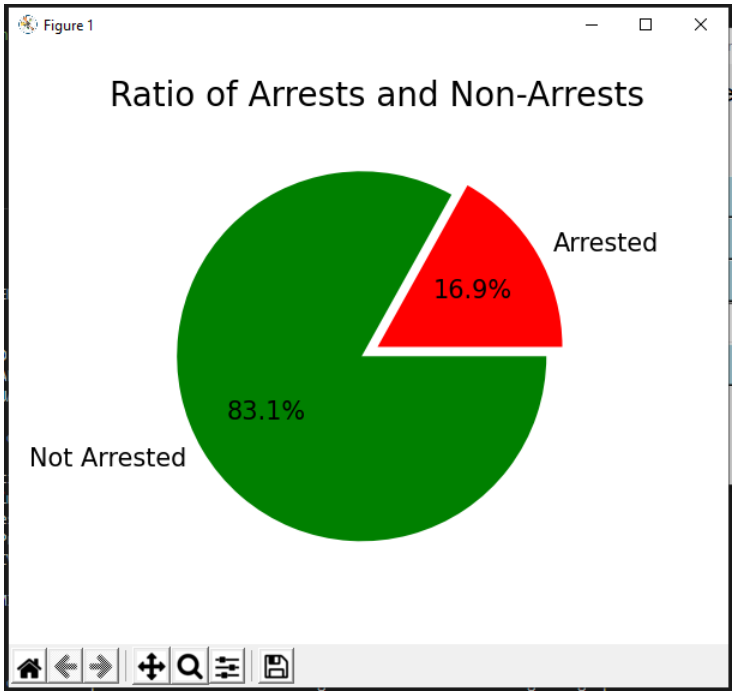




```
Your selection is 3
Chicago crime Data for year 2020 has been successfully fetched from CSV file.
Displaying data of monthly occuring frequency of top5 crimes commited in Chicago in 2020 till date using line graph.
```

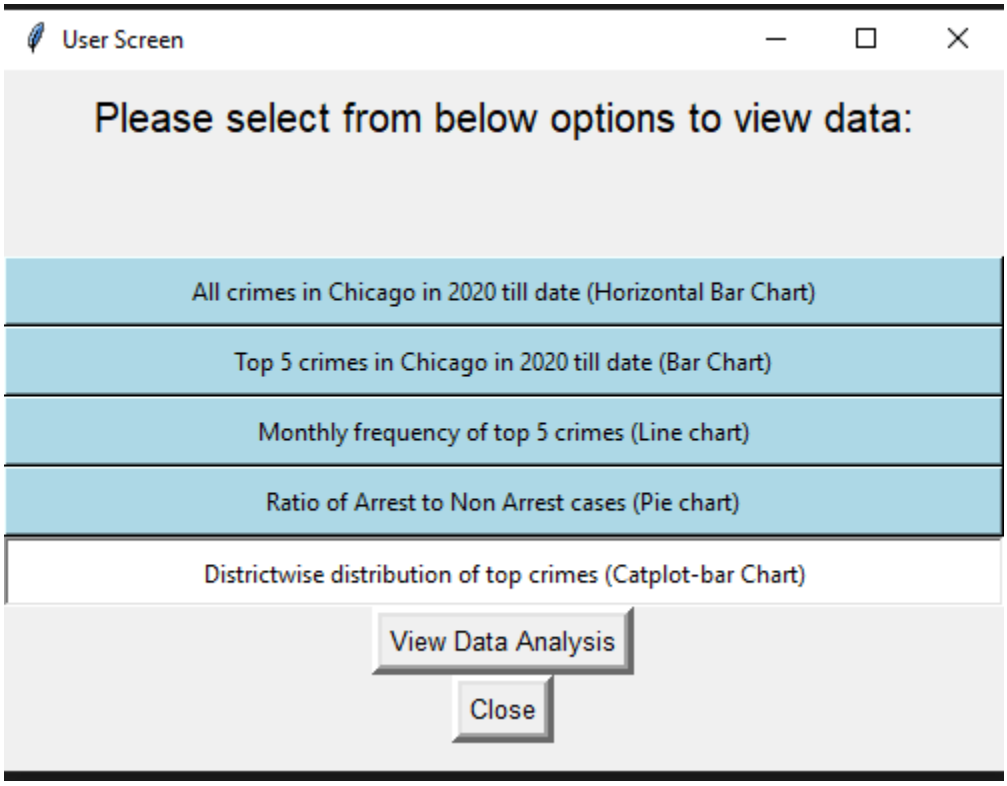
4. Radio Button 4: Ratio of Arrest to Non-Arrest cases (Pie Chart)

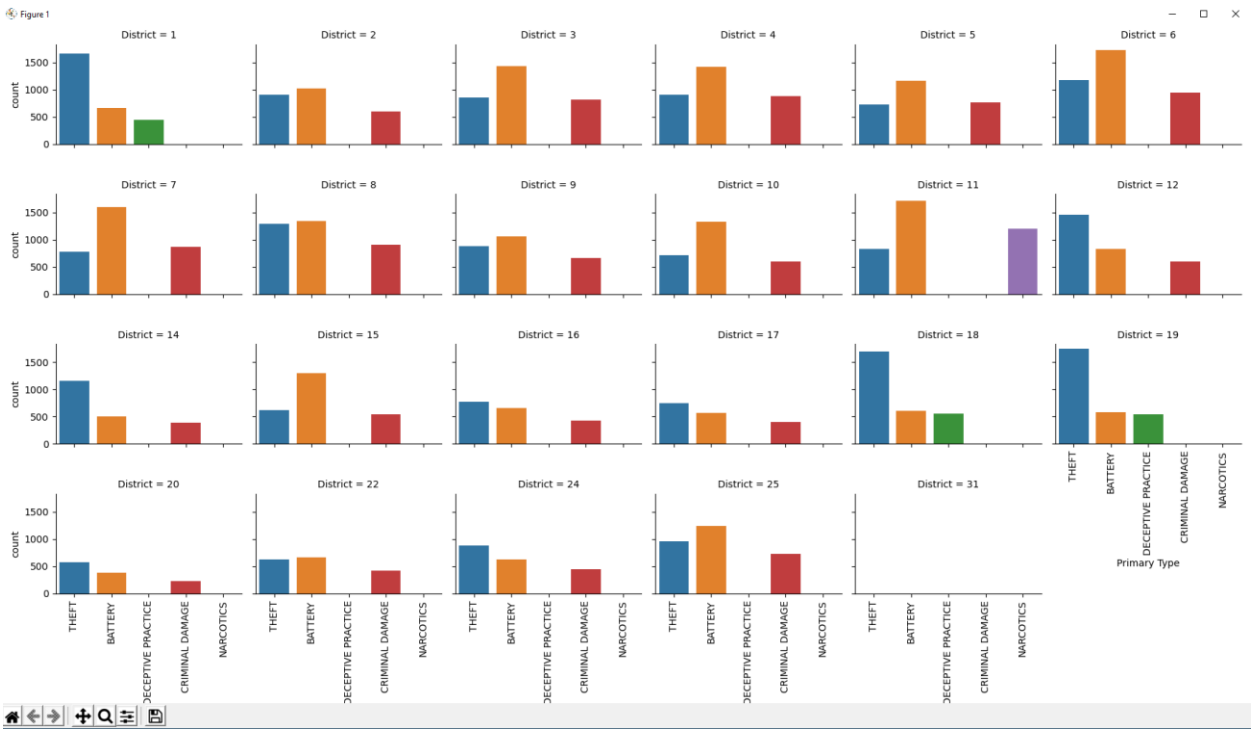




```
Your selection is 4
Chicago crime Data for year 2020 has been successfully fetched from CSV file.
Arrests abd bib-arrests count:
Arrest
True     18342
False    90132
Name: counts, dtype: int64
Displaying data of ratio of arrests vs non arrests to the crimes commited in Chicago in 2020 till date using pie graph.
```

5. Radio Button 5: District wise distribution of top crimes (Catplot-bar chart)

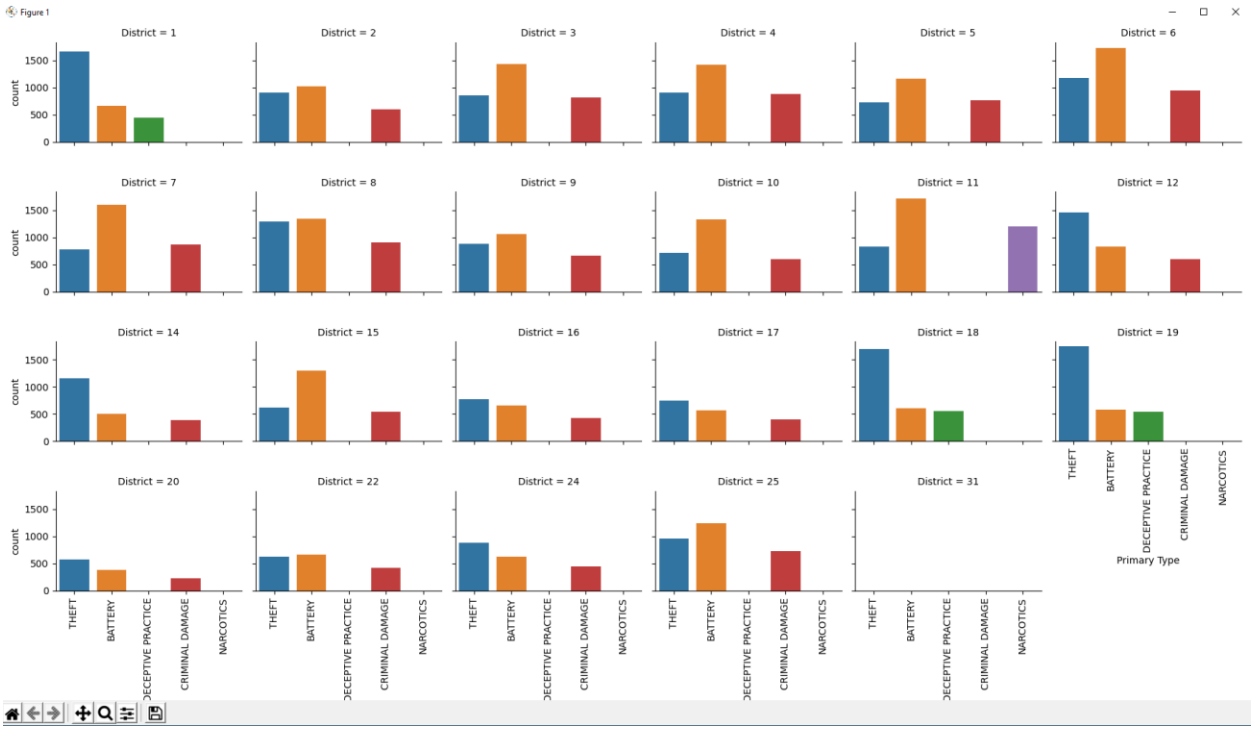





```
Your selection is 5
Chicago crime Data for year 2020 has been successfully fetched from CSV file.
            District       Primary Type  count
District
1         25        1             THEFT   1659
          2         1           BATTERY    664
          9         1  DECEPTIVE PRACTICE    453
2         29        2           BATTERY   1029
          53        2             THEFT    912
...              ...               ...    ...
25        561      25           BATTERY   1243
          584      25             THEFT    958
          565      25    CRIMINAL DAMAGE    732
31        586      31           BATTERY      1
          587      31  DECEPTIVE PRACTICE      1

[68 rows x 3 columns]
Displaying data of districtwise distribution of top crimes commited in Chicago in 2020 till date using catplot bar graph.
```

5. Extra Credits

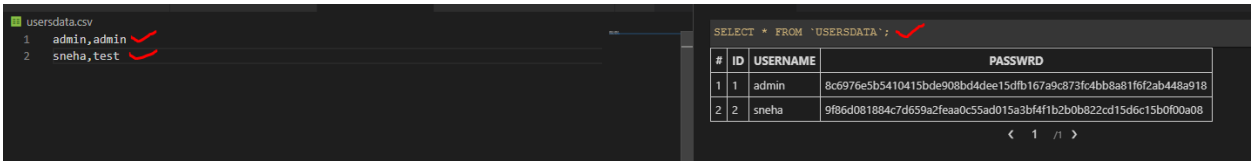
1. Intuitive charting: Side by side graphical depiction

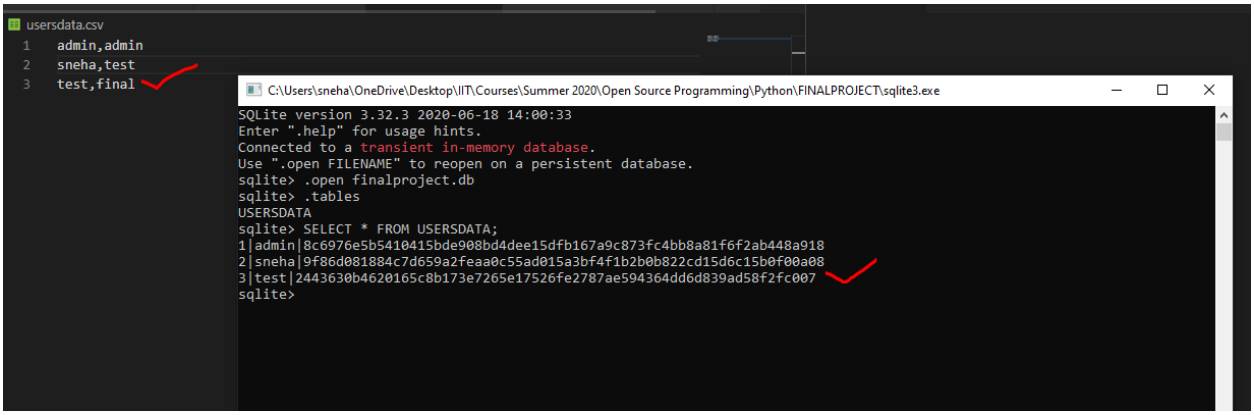


2. Stats on Mean, Standard deviation, variances, counts, averages, correlations etc

	ID	Beat	District	Ward	Community Area	X Coordinate	Y Coordinate	Year	Latitude	Longitude	time_hour	month
count	1.084740e+05	108474.000000	108474.000000	108471.000000	108474.000000	1.077060e+05	1.077060e+05	108474.0	107706.000000	107706.000000	108474.000000	108474.000000
mean	1.197969e+07	1134.628289	11.116710	22.814660	37.668842	1.164940e+06	1.884830e+06	2020.0	41.839562	-87.670282	12.922166	3.664380
std	7.261430e+05	691.827318	6.913048	13.685286	21.439931	1.620089e+04	3.163948e+04	0.0	0.087016	0.058951	6.663892	1.937681
min	2.488900e+04	111.000000	1.000000	1.000000	1.000000	1.092647e+06	1.813897e+06	2020.0	41.644590	-87.934567	0.000000	1.000000
25%	1.198154e+07	611.000000	6.000000	10.000000	23.000000	1.152956e+06	1.857875e+06	2020.0	41.765254	-87.713701	9.000000	2.000000
50%	1.202350e+07	1021.000000	10.000000	23.000000	32.000000	1.166524e+06	1.890014e+06	2020.0	41.853788	-87.664450	14.000000	4.000000
75%	1.206528e+07	1653.000000	16.000000	34.000000	56.000000	1.176651e+06	1.907922e+06	2020.0	41.903113	-87.627522	18.000000	5.000000
max	1.211118e+07	2535.000000	31.000000	50.000000	77.000000	1.205112e+06	1.951527e+06	2020.0	42.022586	-87.524618	23.000000	7.000000

3. SQLite3 DB





4. Jupyter note book

```
#importing statements for libraries
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from datetime import datetime
```

```
#reading the chicago crime data set for year 2020 till date
df = pd.read_csv('Crimes_-_2020.csv')
df.Date = pd.to_datetime(df.Date,format = '%m/%d/%Y %I:%M:%S %p')
df.index = pd.DatetimeIndex(df.Date)
df['time_hour']= df['Date'].apply(lambda x: x.hour) #getting time data
df['month']=df['Date'].apply(lambda x: x.month) #getting month data
df.describe()
```

	ID	Beat	District	Ward	Community Area	X Coordinate	Y Coordinate	Year	Latitude	Longitude	time_hour	month
count	1.084740e+05	108474.000000	108474.000000	108471.000000	108474.000000	1.077060e+05	1.077060e+05	108474.0	107706.000000	107706.000000	108474.000000	108474.000000
mean	1.197969e+07	1134.628289	11.116710	22.814660	37.668842	1.164940e+06	1.884830e+06	2020.0	41.839562	-87.670282	12.922166	3.664380
std	7.261430e+05	691.827318	6.913048	13.685286	21.439931	1.620089e+04	3.163948e+04	0.0	0.087016	0.058951	6.663892	1.937681
min	2.488900e+04	111.000000	1.000000	1.000000	1.000000	1.092647e+06	1.813897e+06	2020.0	41.644590	-87.934567	0.000000	1.000000
25%	1.198154e+07	611.000000	6.000000	10.000000	23.000000	1.152956e+06	1.857875e+06	2020.0	41.765254	-87.713701	9.000000	2.000000
50%	1.202350e+07	1021.000000	10.000000	23.000000	32.000000	1.166524e+06	1.890014e+06	2020.0	41.853788	-87.664450	14.000000	4.000000
75%	1.206528e+07	1653.000000	16.000000	34.000000	56.000000	1.176651e+06	1.907922e+06	2020.0	41.903113	-87.627522	18.000000	5.000000
max	1.211118e+07	2535.000000	31.000000	50.000000	77.000000	1.205112e+06	1.951527e+06	2020.0	42.022586	-87.524618	23.000000	7.000000

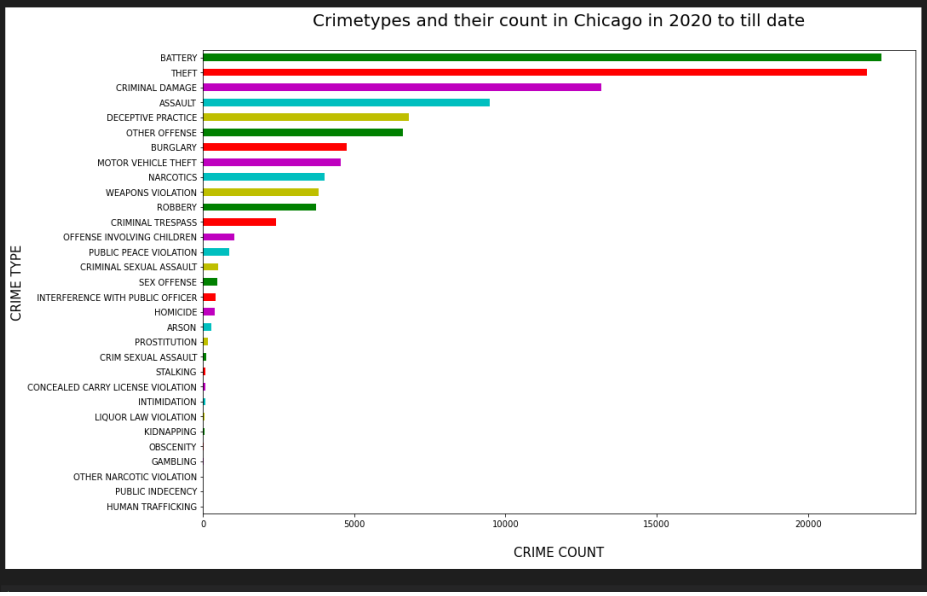
[6]
```
#getting column names and datatypes
df.info()
```

```
3   Block                 108474 non-null  object
4   IUCR                  108474 non-null  object
5   Primary Type          108474 non-null  object
6   Description           108474 non-null  object
7   Location Description  107939 non-null  object
8   Arrest                108474 non-null  bool
9   Domestic              108474 non-null  bool
10  Beat                  108474 non-null  int64
11  District              108474 non-null  int64
12  Ward                  108471 non-null  float64
13  Community Area        108474 non-null  int64
14  FBI Code              108474 non-null  object
15  X Coordinate          107706 non-null  float64
16  Y Coordinate          107706 non-null  float64
17  Year                  108474 non-null  int64
18  Updated On            108474 non-null  object
19  Latitude              107706 non-null  float64
20  Longitude             107706 non-null  float64
21  Location              107706 non-null  object
22  time_hour             108474 non-null  int64
23  month                 108474 non-null  int64
dtypes: bool(2), datetime64[ns](1), float64(5), int64(7), object(9)
memory usage: 15.5+ MB
```

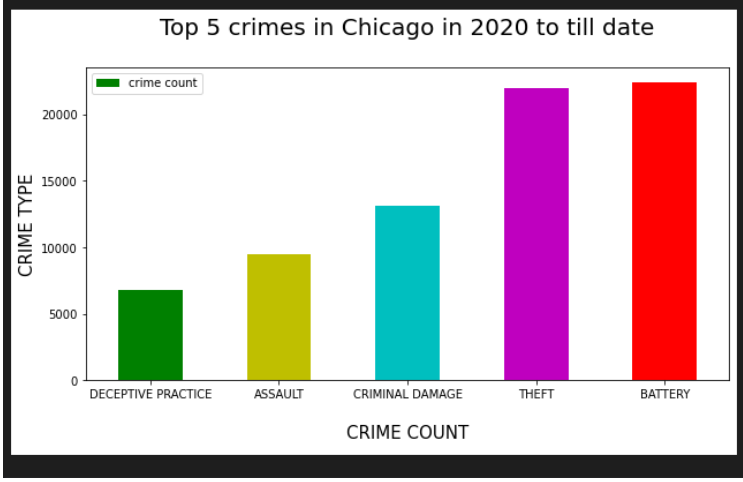
[7]
```
#data of number of crimes in chicago from 2020 to tilldate
crimetypes = df['Primary Type'].unique()
print("Total number of crime types in chicago in year 2020 to till date: ",len(crimetypes))
print("Crimetypes:\n", crimetypes)
```

```
Total number of crime types in chicago in year 2020 to till date:  31
Crimetypes:
 ['BURGLARY' 'MOTOR VEHICLE THEFT' 'THEFT' 'OTHER OFFENSE'
 'CRIMINAL DAMAGE' 'ROBBERY' 'CRIMINAL SEXUAL ASSAULT'
 'DECEPTIVE PRACTICE' 'WEAPONS VIOLATION' 'OFFENSE INVOLVING CHILDREN'
 'HOMICIDE' 'ASSAULT' 'BATTERY' 'NARCOTICS'
 'INTERFERENCE WITH PUBLIC OFFICER' 'SEX OFFENSE' 'INTIMIDATION'
 'STALKING' 'CRIMINAL TRESPASS' 'PUBLIC PEACE VIOLATION' 'ARSON'
 'GAMBLING' 'LIQUOR LAW VIOLATION' 'OBSCENITY'
 'CONCEALED CARRY LICENSE VIOLATION' 'KIDNAPPING' 'PROSTITUTION'
 'HUMAN TRAFFICKING' 'PUBLIC INDECENCY' 'OTHER NARCOTIC VIOLATION'
 'CRIM SEXUAL ASSAULT']
```

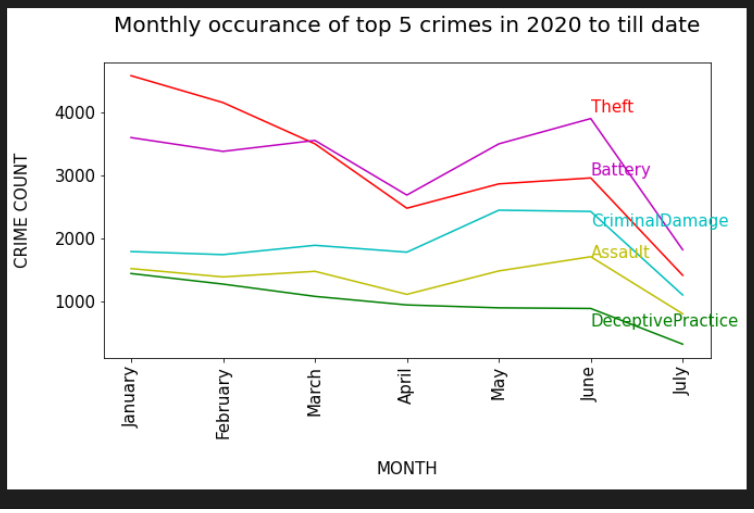
```
#getting all the crimes in chicago in year 2020 to till date
df.groupby(df['Primary Type']).size().sort_values(ascending = True).head(35).plot(kind ='barh',color = ['g', 'y', 'c', 'm', 'r'],figsize=(15,10))
plt.xlabel('\nCRIME COUNT',fontsize='15')
plt.ylabel("CRIME TYPE",fontsize='15')
plt.title("Crimetypes and their count in Chicago in 2020 to till date\n",fontsize="20")
plt.show()
```




```
#getting data of top 5 crimes in chicago in year 2020 to till date
topcrimes = df['Primary Type'].value_counts().sort_values(ascending=False).head()
tp = df.groupby('Primary Type', as_index=False).agg({"ID": "count"})
tp = tp.sort_values(by=['ID'], ascending=False).head()
tp = tp.sort_values(by='ID', ascending=True)
print("Top 5 crimes and counts: ")
display(tp) #displaying as table
print()
colors = ['g', 'y', 'c', 'm', 'r']
tp.plot.bar(x='Primary Type',y='ID',color = colors,label ='crime count',figsize=(10,5))
plt.title("Top 5 crimes in Chicago in 2020 to till date\n",fontsize='20')
plt.xlabel("\nCRIME COUNT",fontsize='15')
plt.ylabel("CRIME TYPE",fontsize='15')
plt.xticks(rotation=360)
plt.show()
```



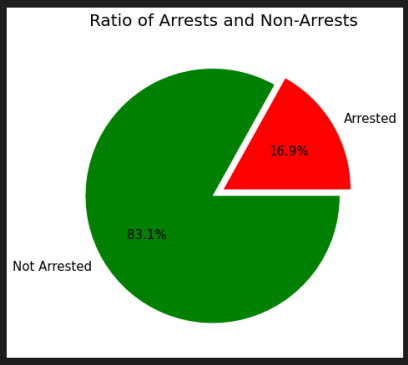
```
plt.text(5,1700,"Assault",fontsize=15,color="y")
plt.text(5,600,"DeceptivePractice",fontsize=15,color="g")

p.set_title("Monthly occurance of top 5 crimes in 2020 to till date\n", fontsize=20)
p.set_xlabel("\nMONTH", fontsize=15)
p.set_ylabel("CRIME COUNT\n", fontsize=15)

plt.show()
```



[16]

```
#get ratio of arrest count to non-arrest count
ar = df[['Arrest']]  # get a series from data frame
arrests = pd.DataFrame(ar.groupby('Arrest').size().sort_values(ascending=True).rename('counts'))
display(arrests)
explode=[0,0.10]
plt.figure(figsize=(7,7))
plt.pie(arrests,labels=["Arrested","Not Arrested"], autopct="%.1f%%", explode=explode, colors=["r","g"],textprops={'fontsize': 15})
plt.title("Ratio of Arrests and Non-Arrests",fontsize=20)
plt.show()
```

Arrest	counts
True	18342
False	90132

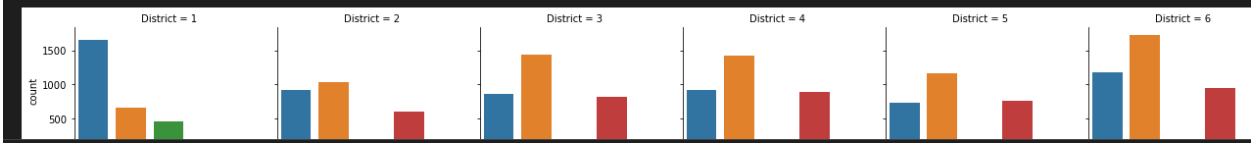


[95]

```
#get data of district wise top crimes
districtcrimes = df.groupby(['District', 'Primary Type']).size().reset_index(name='count').groupby('District').apply(lambda x: x.sort_values('count',ascending=F
display(districtcrimes)
#Creating multiple plots
data =sns.catplot("Primary Type",y='count', col="District", col_wrap=6, data=districtcrimes, kind='bar',height=3)
for x in data.axes:
    plt.setp(x.get_xticklabels(), rotation=90)
plt.tight_layout()
#plt.title("Districtwise distribution of crimes in Chicago 2020 till date")
plt.subplots_adjust(hspace=0.5)
```

		District	Primary Type	count
District				
1	25	1	THEFT	1659
		2	1	BATTERY	664
		9	1	DECEPTIVE PRACTICE	453
2	29	2	BATTERY	1029
		53	2	THEFT	912
...	...	...	...	...	...
25	561	25	BATTERY	1243
		584	25	THEFT	958
		565	25	CRIMINAL DAMAGE	732
31	586	31	BATTERY	1
		587	31	DECEPTIVE PRACTICE	1

68 rows x 3 columns



6. Future Work

- Perform predictive analysis on crime events.
- Gather Crime vs Time frequency
- Yearly increase in crimes with large dataset file

7. Conclusion

Through the analysis of crime data, we were able to find out a few answers regarding the crimes in Chicago. Few of the answers include, most committed crimes in Chicago in 2020 and found that no arrests were made in 83% of the crimes. This project has a great deal of scope and with future work, predictive models can be built.