PROJECT: ANALYZING CRIME IN LOS ANGELES

Los Angeles, California 😎. The City of Angels. Tinseltown. The Entertainment Capital of the World!

Known for its warm weather, palm trees, sprawling coastline, and Hollywood, along with producing some of the most iconic films and songs. However, as with any highly populated city, it isn't always glamorous and there can be a large volume of crime. That's where you can help!

You have been asked to support the Los Angeles Police Department (LAPD) by analyzing crime data to identify patterns in criminal behavior. They plan to use your insights to allocate resources effectively to tackle various crimes in different areas.

The Data

They have provided you with a single dataset to use. A summary and preview are provided below.

It is a modified version of the original data, which is publicly available from Los Angeles Open Data.

crimes.csv

Column	Description
`'DR_NO'`	Division of Records Number: Official file number made up of a 2-digit year, area ID, and 5 digits.
`'Date Rptd'`	Date reported - MM/DD/YYYY.
`'DATE OCC'`	Date of occurrence - MM/DD/YYYY.
`'TIME OCC'`	In 24-hour military time.
`'AREA NAME'`	The 21 Geographic Areas or Patrol Divisions are also given a name designation that references a landmark or the surrounding community that it is responsible for. For example, the 77th Street Division is located at the intersection of South Broadway and 77th Street, serving neighborhoods in South Los Angeles.
`'Crm Cd Desc'`	Indicates the crime committed.
`'Vict Age'`	Victim's age in years.
`'Vict Sex'`	Victim's sex: `F`: Female, `M`: Male, `X`: Unknown.
`'Vict Descent'`	Victim's descent: • `A` - Other Asian • `B` - Black • `C` - Chinese • `D` - Cambodian • `F` - Filipino • `G` - Guamanian • `H` - Hispanic/Latin/Mexican • `I` - American Indian/Alaskan Native • `J` - Japanese • `K` - Korean • `L` - Laotian • `O` - Other • `P` - Pacific Islander • `S` - Samoan • `U` - Hawaiian

Column	Description
'LOCATION'	Street address of the crime.

```
# Re-run this cell
# Import required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
crimes = pd.read_csv("crimes.csv", dtype={"TIME OCC": str})
crimes.head()
```

	D	D…	D…		AR…	Crm Cd Desc			Vict …	We…
0	220314085	2022-07-22	2020-05-12	1110	Southwest	THEFT OF IDENTITY	27	F	B	null
1	222013040	2022-08-06	2020-06-04	1620	Olympic	THEFT OF IDENTITY	60	M	H	null
2	220614831	2022-08-18	2020-08-17	1200	Hollywood	THEFT OF IDENTITY	28	M	H	null
3	231207725	2023-02-27	2020-01-27	0635	77th Street	THEFT OF IDENTITY	37	M	H	null
4	220213256	2022-07-14	2020-07-14	0900	Rampart	THEFT OF IDENTITY	79	M	B	null

Rows: 5 Expand Table

```
# Convert 'TIME OCC' to integer
crimes['TIME OCC'] = crimes['TIME OCC'].astype(int)

# Highest frequency of crimes
crimes['HOUR'] = crimes["TIME OCC"] // 100


# Find the peak crime hour
peak_crime_hour = crimes["HOUR"].value_counts().idxmax()

print(peak_crime_hour)
```

```
12
```

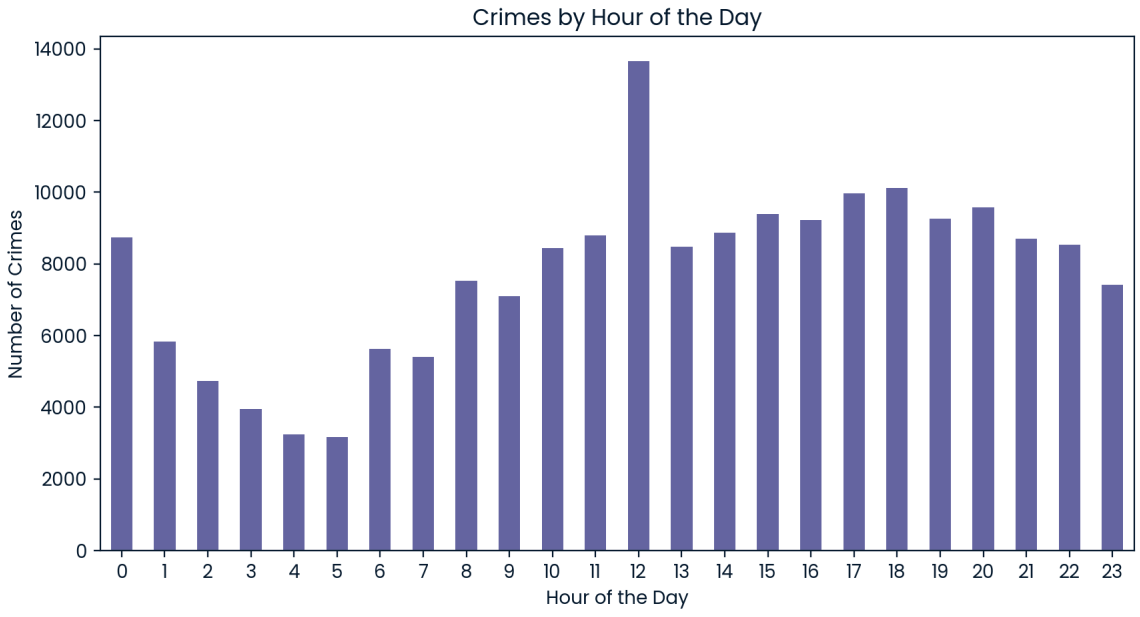
```
# Filter for night crimes using full time values
night_crimes = crimes[(crimes['TIME OCC'] >= 2200) | (crimes['TIME OCC'] <= 359)]

# Find the area with the highest number of night crimes
peak_night_crime_location = night_crimes['AREA NAME'].value_counts().idxmax()

print(peak_night_crime_location)
```

```
Central
```

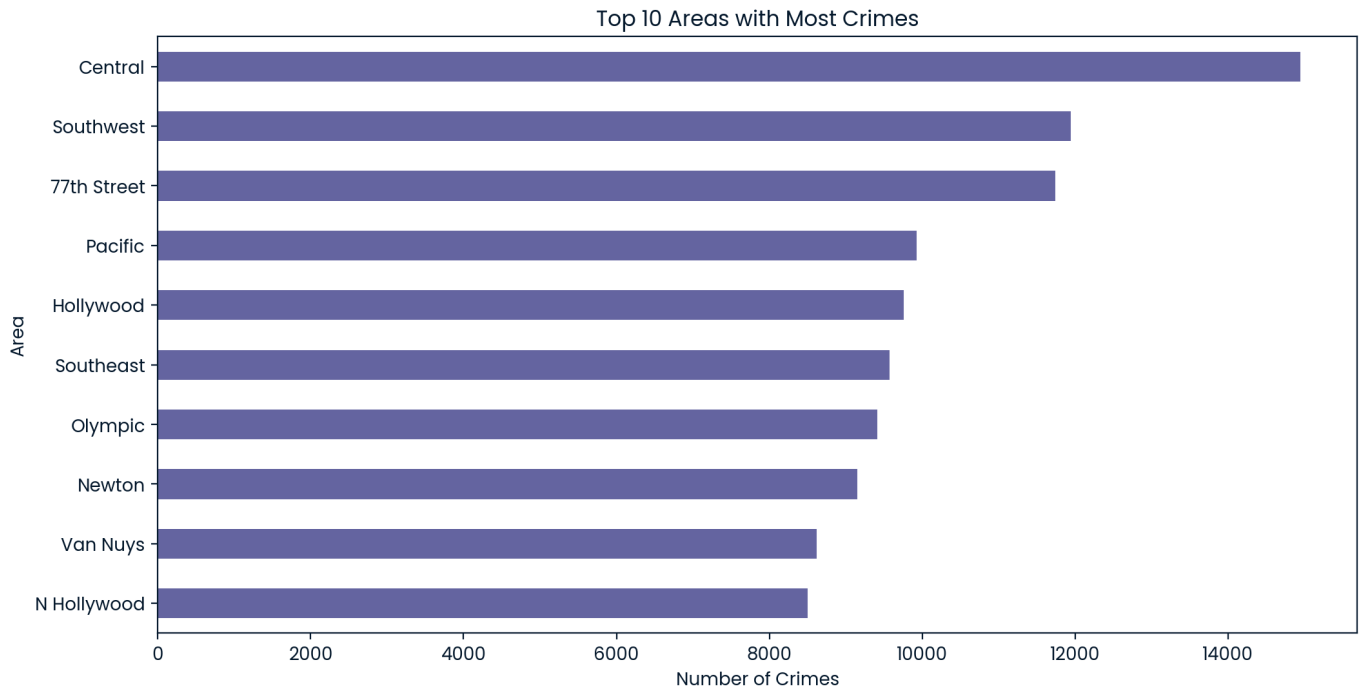
```
# Define the bins and labels
bins = [0, 17, 25, 34, 44, 54, 64, crimes['Vict Age'].max()]
labels = ["0-17", "18-25", "26-34", "35-44", "45-54", "55-64", "65+"]

# Create a new column for age groups
crimes['AGE_GROUP'] = pd.cut(crimes['Vict Age'], bins=bins, labels=labels, right=True)

# Count number of crimes in each age group
victim_ages = crimes['AGE_GROUP'].value_counts().sort_index()

print(victim_ages)
```

```
0-17      4528
18-25    28291
26-34    47470
35-44    42157
45-54    28353
55-64    20169
65+      14747
Name: AGE_GROUP, dtype: int64
```

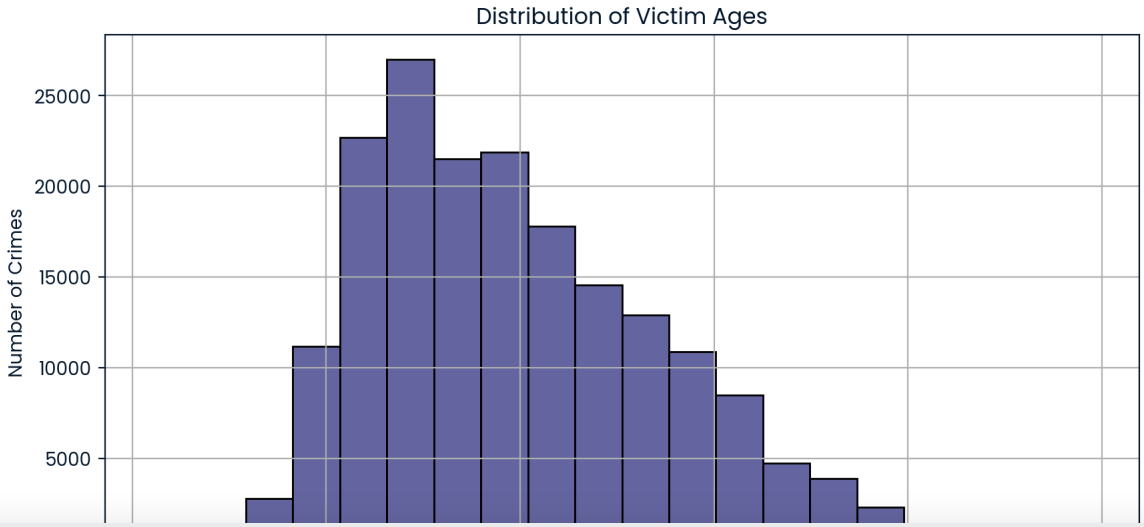
```
plt.figure(figsize=(10, 5))
crimes['HOUR'].value_counts().sort_index().plot(kind='bar')
plt.title('Crimes by Hour of the Day')
plt.xlabel('Hour of the Day')
plt.ylabel('Number of Crimes')
plt.xticks(rotation=0)
plt.show()
```



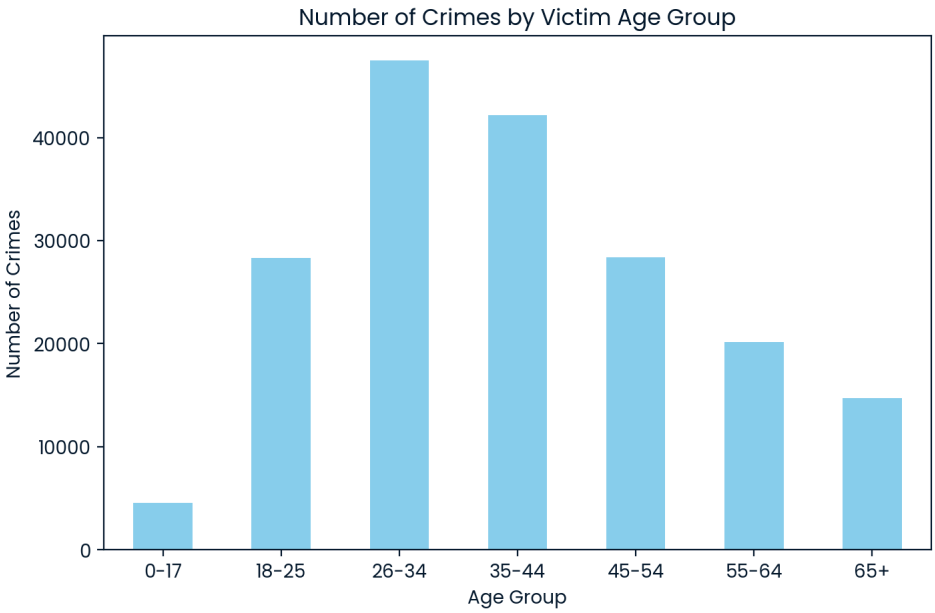
```
plt.figure(figsize=(12, 6))
crimes['AREA NAME'].value_counts().head(10).plot(kind='barh')
plt.title('Top 10 Areas with Most Crimes')
plt.xlabel('Number of Crimes')
plt.ylabel('Area')
plt.gca().invert_yaxis()
plt.show()
```



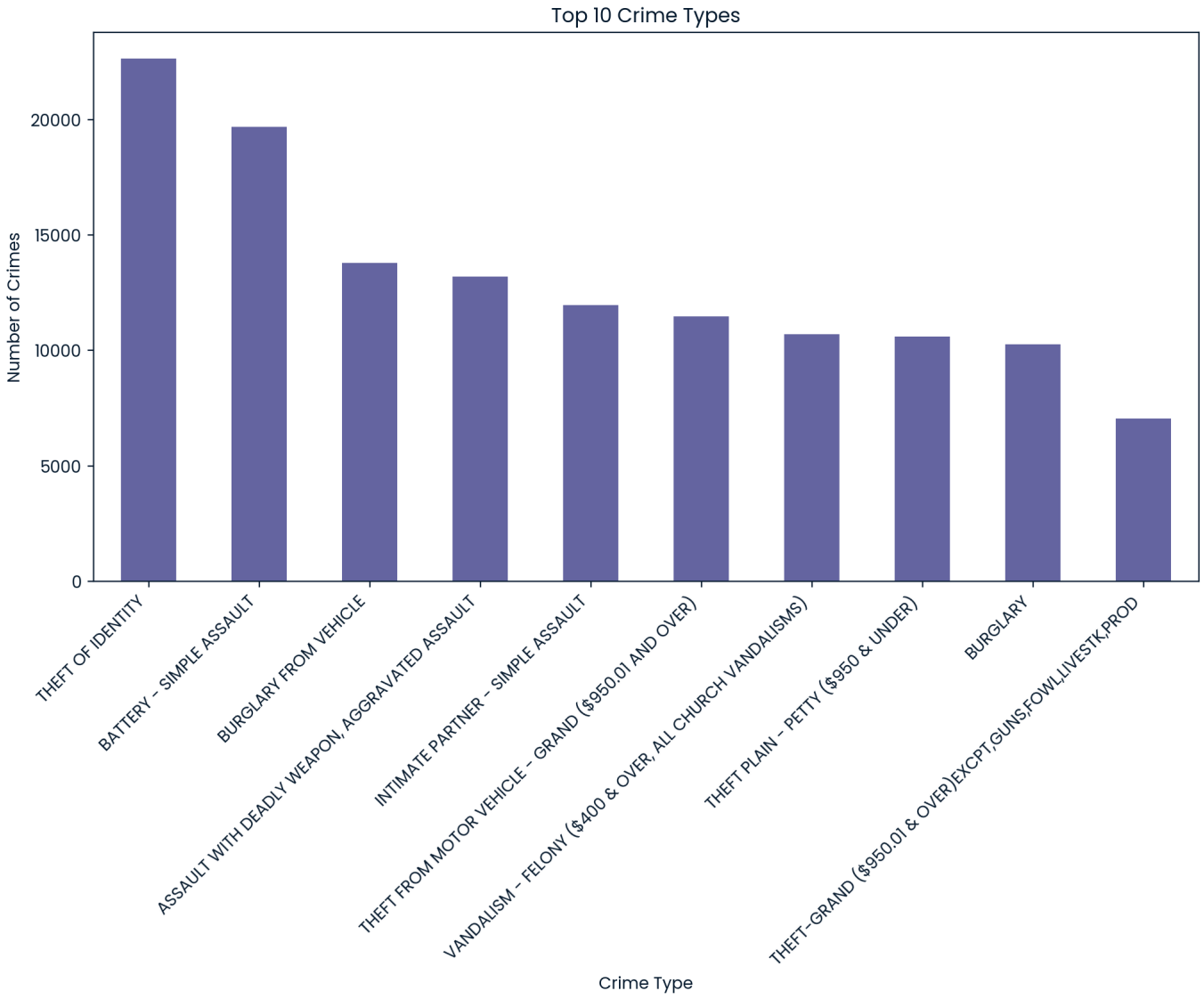

```
plt.figure(figsize=(10, 5))
crimes['Vict Age'].hist(bins=20, edgecolor='black')
plt.title('Distribution of Victim Ages')
plt.xlabel('Age')
plt.ylabel('Number of Crimes')
plt.show()
```



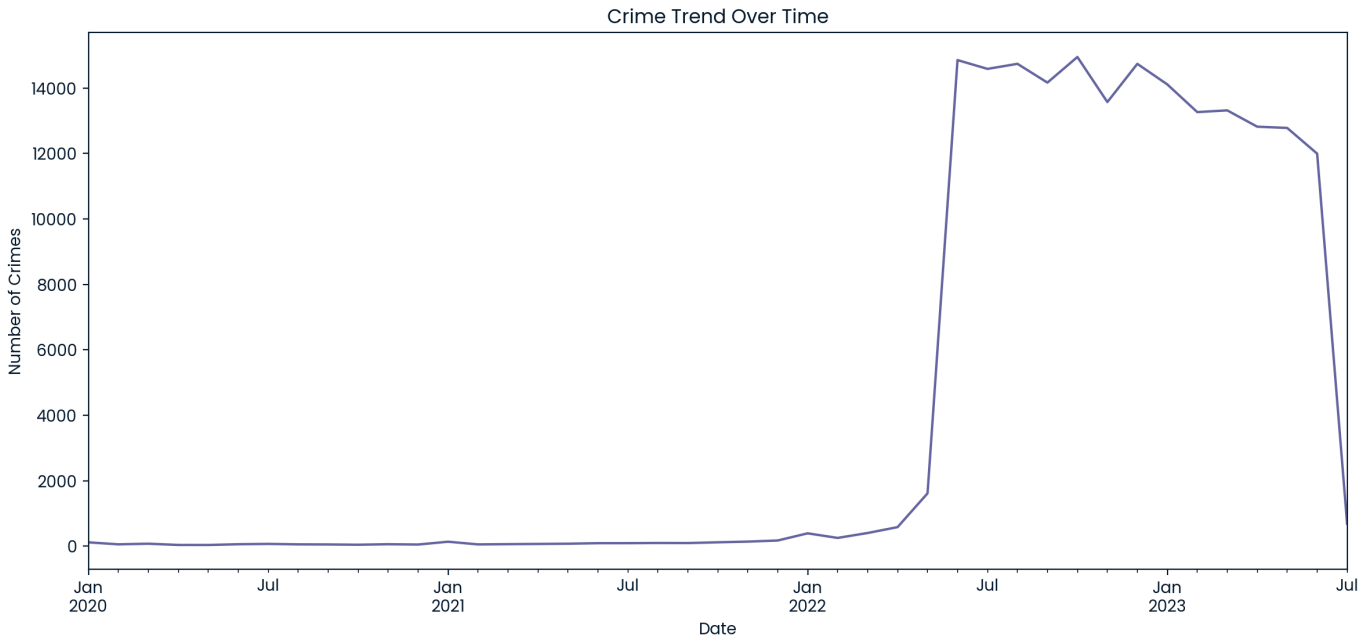
```
plt.figure(figsize=(8, 5))
victim_ages.plot(kind='bar', color='skyblue')
plt.title('Number of Crimes by Victim Age Group')
plt.xlabel('Age Group')
plt.ylabel('Number of Crimes')
plt.xticks(rotation=0)
plt.show()
```



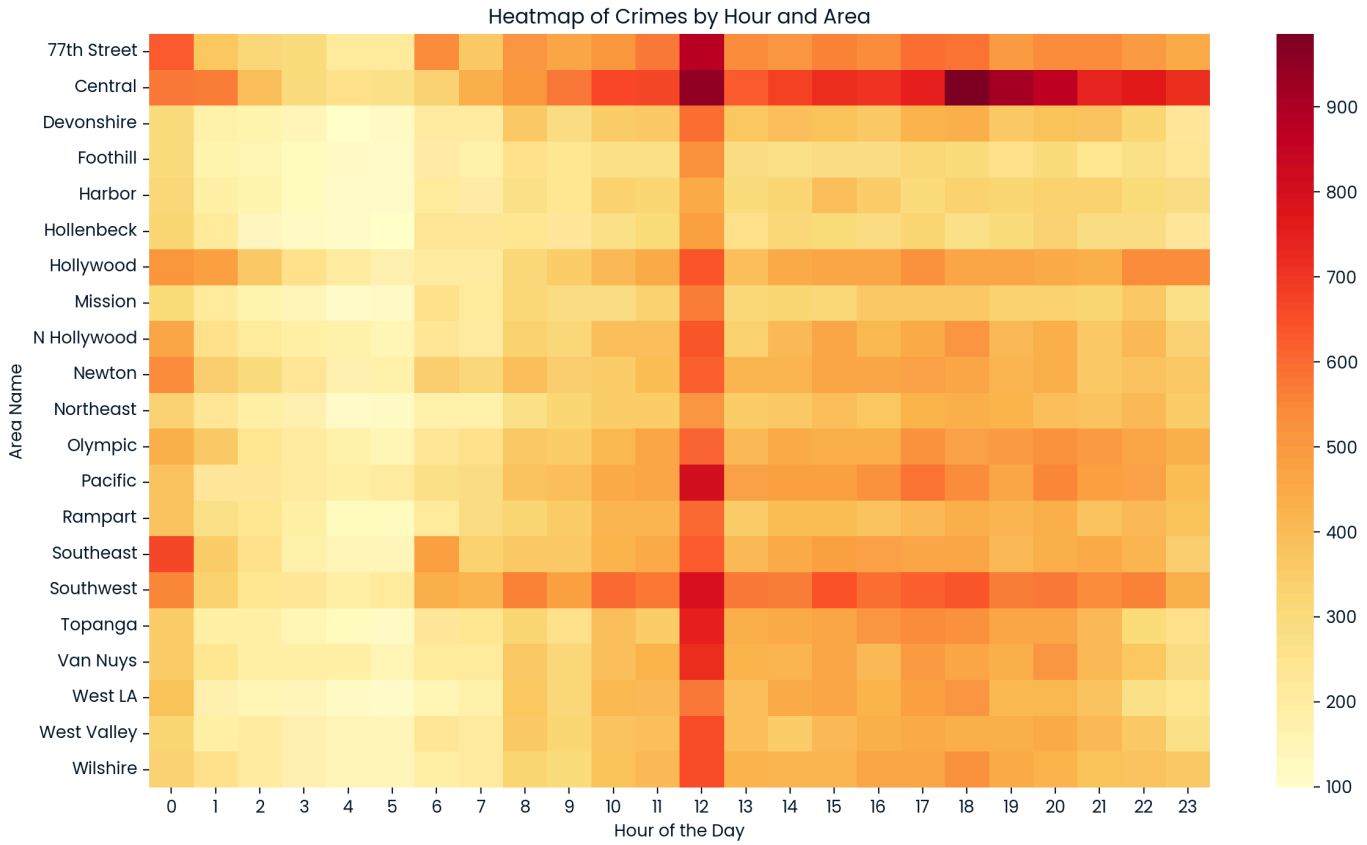
```
plt.figure(figsize=(12, 6))
crimes['Crm Cd Desc'].value_counts().head(10).plot(kind='bar')
plt.title('Top 10 Crime Types')
plt.xlabel('Crime Type')
plt.ylabel('Number of Crimes')
plt.xticks(rotation=45, ha='right')
plt.show()
```



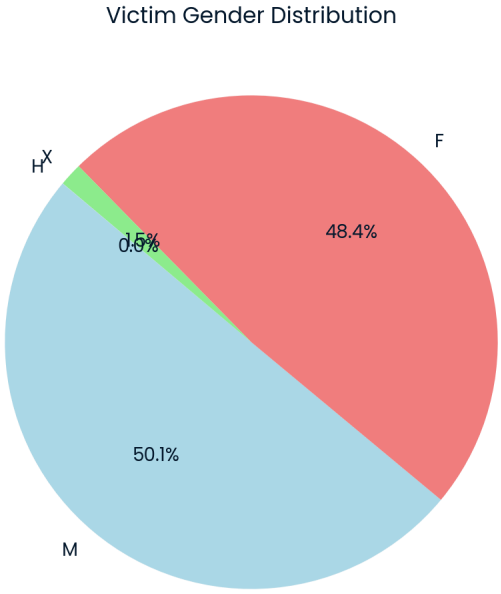
```
# Convert 'DATE OCC' to datetime
crimes['DATE OCC'] = pd.to_datetime(crimes['DATE OCC'])

plt.figure(figsize=(14, 6))
crimes.groupby(crimes['DATE OCC'].dt.to_period('M')).size().plot(kind='line')
plt.title('Crime Trend Over Time')
plt.xlabel('Date')
plt.ylabel('Number of Crimes')
plt.show()
```



```
pivot = crimes.pivot_table(index='AREA NAME', columns='HOUR', values='DR_NO', aggfunc='count')

plt.figure(figsize=(14, 8))
sns.heatmap(pivot, cmap='YlOrRd')
plt.title('Heatmap of Crimes by Hour and Area')
plt.xlabel('Hour of the Day')
plt.ylabel('Area Name')
plt.show()
```



```
plt.figure(figsize=(6, 6))
crimes['Vict Sex'].value_counts().plot(kind='pie', autopct='%1.1f%%', startangle=140, colors=['lightblue', 'lightcoral',
'lightgreen'])
plt.title('Victim Gender Distribution')
plt.ylabel('')  # Hide y-label
plt.show()
```



```
# Convert date to datetime if not already
crimes['DATE OCC'] = pd.to_datetime(crimes['DATE OCC'])
crimes['DAY_OF_WEEK'] = crimes['DATE OCC'].dt.day_name()

plt.figure(figsize=(10, 5))
crimes['DAY_OF_WEEK'].value_counts().reindex(['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday',
'Sunday']).plot(kind='bar', color='coral')
plt.title('Crimes by Day of the Week')
plt.xlabel('Day of the Week')
plt.ylabel('Number of Crimes')
plt.xticks(rotation=45)
plt.show()
```