```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime
import plotly.express as px
from sklearn.preprocessing import LabelEncoder
from scipy.stats import pearsonr



df = pd.read_csv('/content/CRIME.csv')



print(df.head(10))
```

```
   Report Number     Date Reported Date of Occurrence Time of Occurrence  \
0              1  02-01-2020 00:00   01-01-2020 00:00   01-01-2020 01:11
1              2  01-01-2020 19:00   01-01-2020 01:00   01-01-2020 06:26
2              3  02-01-2020 05:00   01-01-2020 02:00   01-01-2020 14:30
3              4  01-01-2020 05:00   01-01-2020 03:00   01-01-2020 14:46
4              5  01-01-2020 21:00   01-01-2020 04:00   01-01-2020 16:51
5              6  02-01-2020 03:00   01-01-2020 05:00   01-01-2020 17:09
6              7  01-01-2020 16:00   01-01-2020 06:00   01-01-2020 14:08
7              8  02-01-2020 10:00   01-01-2020 07:00   02-01-2020 06:33
8              9  04-01-2020 03:00   01-01-2020 08:00   02-01-2020 06:34
9             10  03-01-2020 07:00   01-01-2020 09:00   01-01-2020 17:50

        City  Crime Code    Crime Description  Victim Age Victim Gender  \
0  Ahmedabad         576       IDENTITY THEFT          16             M
1    Chennai         128             HOMICIDE          37             M
2   Ludhiana         271           KIDNAPPING          48             F
3       Pune         170             BURGLARY          49             F
4       Pune         421            VANDALISM          30             F
5      Delhi         442              ASSAULT          16             M
6    Chennai         172     VEHICLE - STOLEN          64             F
7    Chennai         169       COUNTERFEITING          78             X
8     Mumbai         338            EXTORTION          41             X
9    Chennai         497  PUBLIC INTOXICATION          29             M

    Weapon Used   Crime Domain  Police Deployed Case Closed  Date Case Closed
0  Blunt Object  Violent Crime               13          No               NaN
1        Poison    Other Crime                9          No               NaN
2  Blunt Object    Other Crime               15          No               NaN
3       Firearm    Other Crime                1         Yes  29-04-2020 05:00
4         Other    Other Crime               18         Yes  08-01-2020 21:00
5       Firearm  Violent Crime               18         Yes  30-03-2020 03:00
6         Knife  Violent Crime               13         Yes  24-03-2020 16:00
```

```
7         Knife    Other Crime               8          No              NaN
8  Blunt Object    Other Crime               1          No              NaN
9         Knife    Other Crime               4          No              NaN
```

Research crime statistics (from 2014 onwards) in Tamil Nadu

```
tn_df = df[df['City'].isin(['Chennai'])]
```

```
print(tn_df)
```

```
        Report Number     Date Reported Date of Occurrence Time of Occurrence  \
1                   2  01-01-2020 19:00   01-01-2020 01:00   01-01-2020 06:26
6                   7  01-01-2020 16:00   01-01-2020 06:00   01-01-2020 14:08
7                   8  02-01-2020 10:00   01-01-2020 07:00   02-01-2020 06:33
9                  10  03-01-2020 07:00   01-01-2020 09:00   01-01-2020 17:50
23                 24  03-01-2020 09:00   01-01-2020 23:00   01-01-2020 23:40
...               ...               ...               ...               ...
40080           40081  28-07-2024 19:00   07-28-2024 00:00   28-07-2024 03:39
40086           40087  29-07-2024 23:00   07-28-2024 06:00   28-07-2024 09:53
40116           40117  30-07-2024 01:00   07-29-2024 12:00   29-07-2024 13:43
40125           40126  01-08-2024 02:00   07-29-2024 21:00   30-07-2024 05:05
40154           40155  01-08-2024 11:00   07-31-2024 02:00   31-07-2024 12:59

           City  Crime Code    Crime Description  Victim Age Victim Gender  \
1       Chennai         128            HOMICIDE          37             M
6       Chennai         172    VEHICLE - STOLEN          64             F
7       Chennai         169      COUNTERFEITING          78             X
9       Chennai         497  PUBLIC INTOXICATION          29             M
23      Chennai         322           VANDALISM          77             F
...         ...         ...                 ...         ...           ...
40080   Chennai         341           VANDALISM          71             F
40086   Chennai         253             ROBBERY          73             F
40116   Chennai         341           EXTORTION          25             F
40125   Chennai         591          KIDNAPPING          18             F
40154   Chennai         532      COUNTERFEITING          50             F

         Weapon Used   Crime Domain  Police Deployed Case Closed  \
1             Poison    Other Crime                9          No
6              Knife  Violent Crime               13         Yes
7              Knife    Other Crime                8          No
9              Knife    Other Crime                4          No
23             Other    Other Crime               18          No
...              ...            ...              ...         ...
40080         Poison    Other Crime               10          No
40086          Other  Violent Crime               14          No
40116  Blunt Object    Other Crime                4         Yes
40125           NaN    Other Crime                9         Yes
```

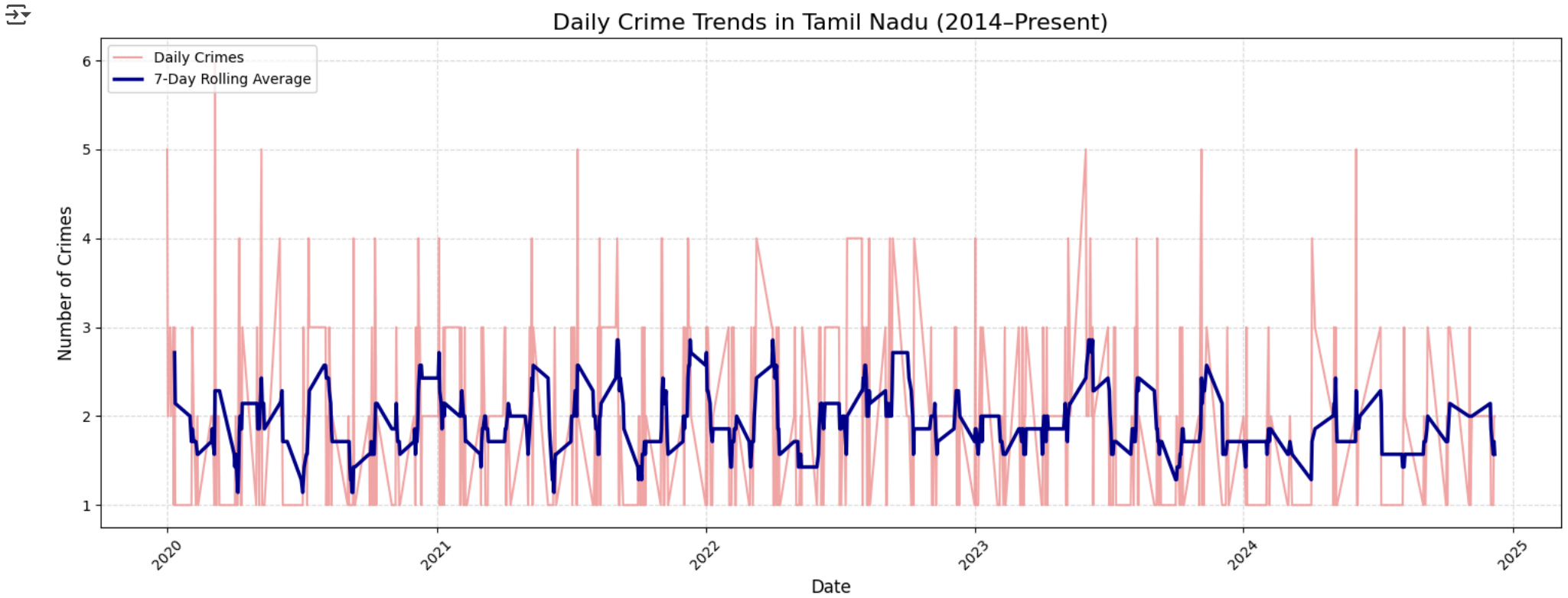
```
40154    Explosives    Other Crime               2          No

       Date Case Closed
1                   NaN
6      24-03-2020 16:00
7                   NaN
9                   NaN
23                  NaN
...                 ...
40080               NaN
40086               NaN
40116  12-10-2024 01:00
40125  27-08-2024 02:00
40154               NaN

[2493 rows x 14 columns]
```

Analyze trends over time

```
df['Date Reported'] = pd.to_datetime(df['Date Reported'], dayfirst=True, errors='coerce')
df['Date of Occurrence'] = pd.to_datetime(df['Date of Occurrence'], dayfirst=True, errors='coerce')
df['Time of Occurrence'] = pd.to_datetime(df['Time of Occurrence'], dayfirst=True, errors='coerce')
df['Date Case Closed'] = pd.to_datetime(df['Date Case Closed'], dayfirst=True, errors='coerce')

df['Hour of Day'] = df['Time of Occurrence'].dt.hour
df['Day of Week'] = df['Date of Occurrence'].dt.day_name()
df['Month'] = df['Date of Occurrence'].dt.month_name()
df['Year'] = df['Date of Occurrence'].dt.year
```

```
import matplotlib.pyplot as plt

daily_counts = tn_df.groupby(tn_df['Date of Occurrence'].dt.date).size()

rolling_avg = daily_counts.rolling(window=7).mean()

plt.figure(figsize=(15, 6))
plt.plot(daily_counts.index, daily_counts.values, label='Daily Crimes', color='lightcoral', linewidth=1.5, alpha=0.7)
plt.plot(rolling_avg.index, rolling_avg.values, label='7-Day Rolling Average', color='darkblue', linewidth=2.5)

plt.title('Daily Crime Trends in Tamil Nadu (2014–Present)', fontsize=16)
plt.xlabel('Date', fontsize=12)
plt.ylabel('Number of Crimes', fontsize=12)
plt.xticks(rotation=45)
plt.grid(True, linestyle='--', alpha=0.4)
plt.legend(loc='upper left')
```

```
plt.tight_layout()
plt.savefig('daily_crime_trends_lineplot.png')
plt.show()
```

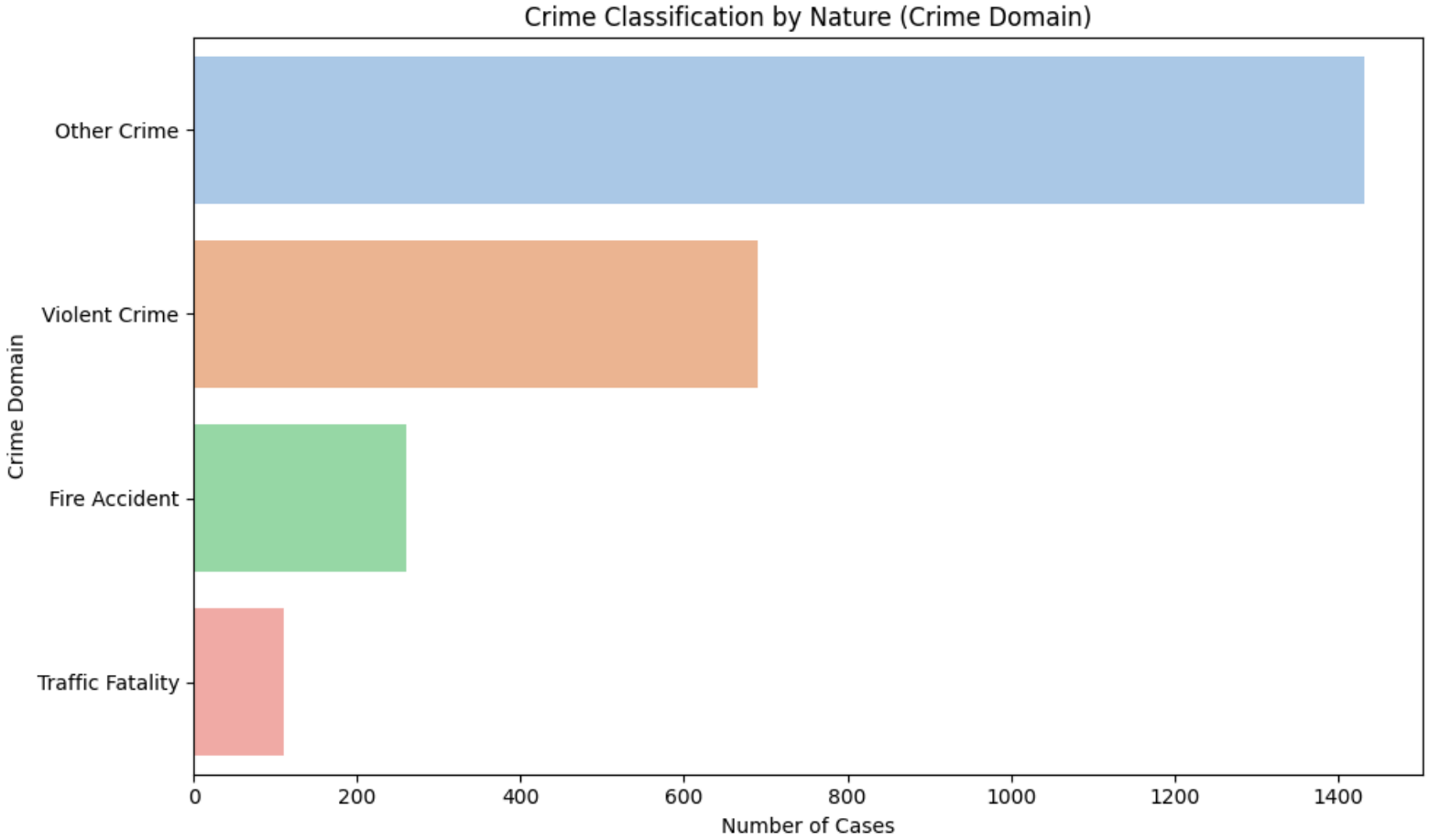


Classify crimes based on their nature.

```
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
sns.countplot(y='Crime Domain', data=tn_df, order=tn_df['Crime Domain'].value_counts().index, palette='pastel')
plt.title('Crime Classification by Nature (Crime Domain)')
plt.xlabel('Number of Cases')
plt.ylabel('Crime Domain')
plt.tight_layout()
plt.savefig('crime_classification_domain.png')
```

```
plt.show()
```

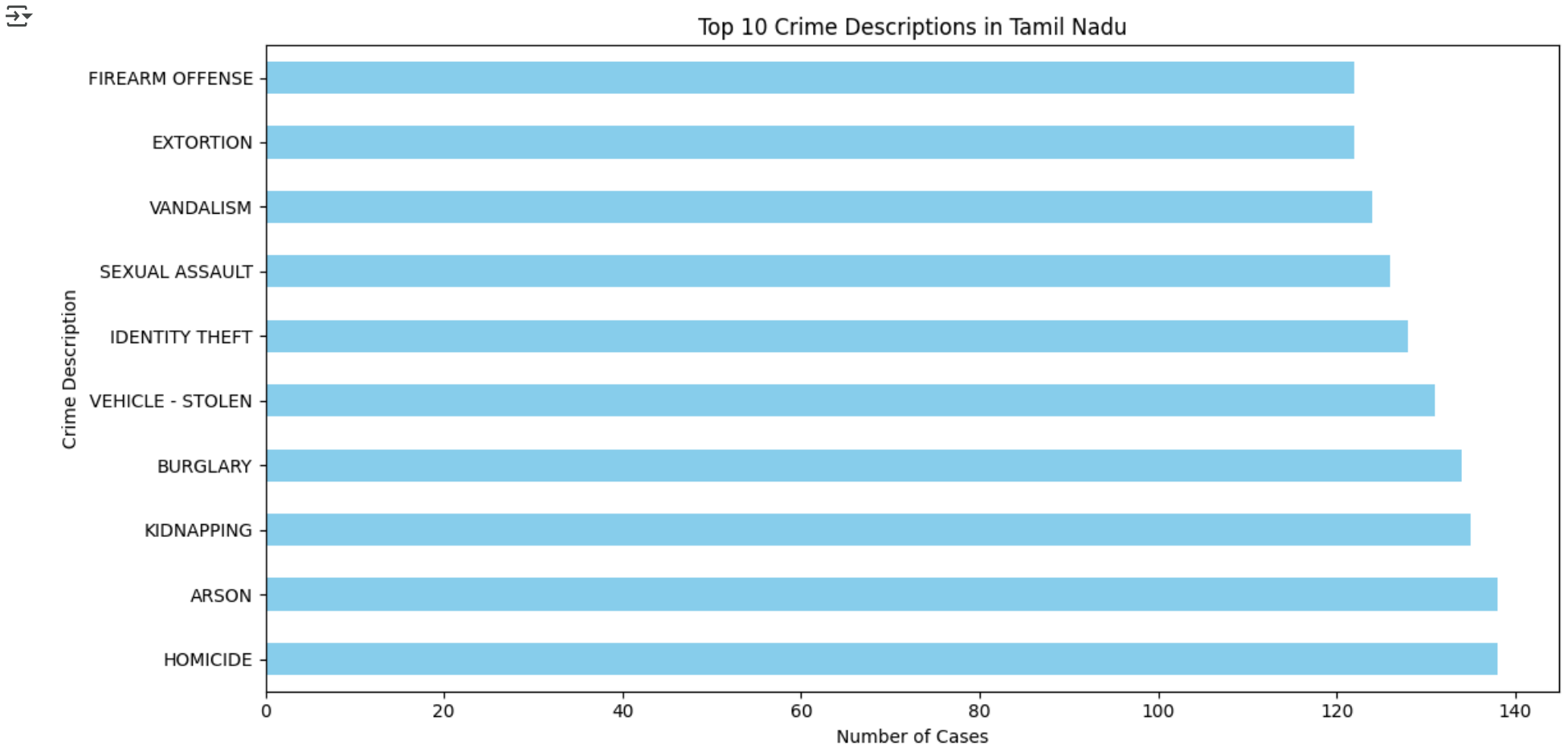
```
/tmp/ipython-input-12-284429932.py:5: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for t

  sns.countplot(y='Crime Domain', data=tn_df, order=tn_df['Crime Domain'].value_counts().index, palette='pastel')
```



Generate innovative statistical charts.

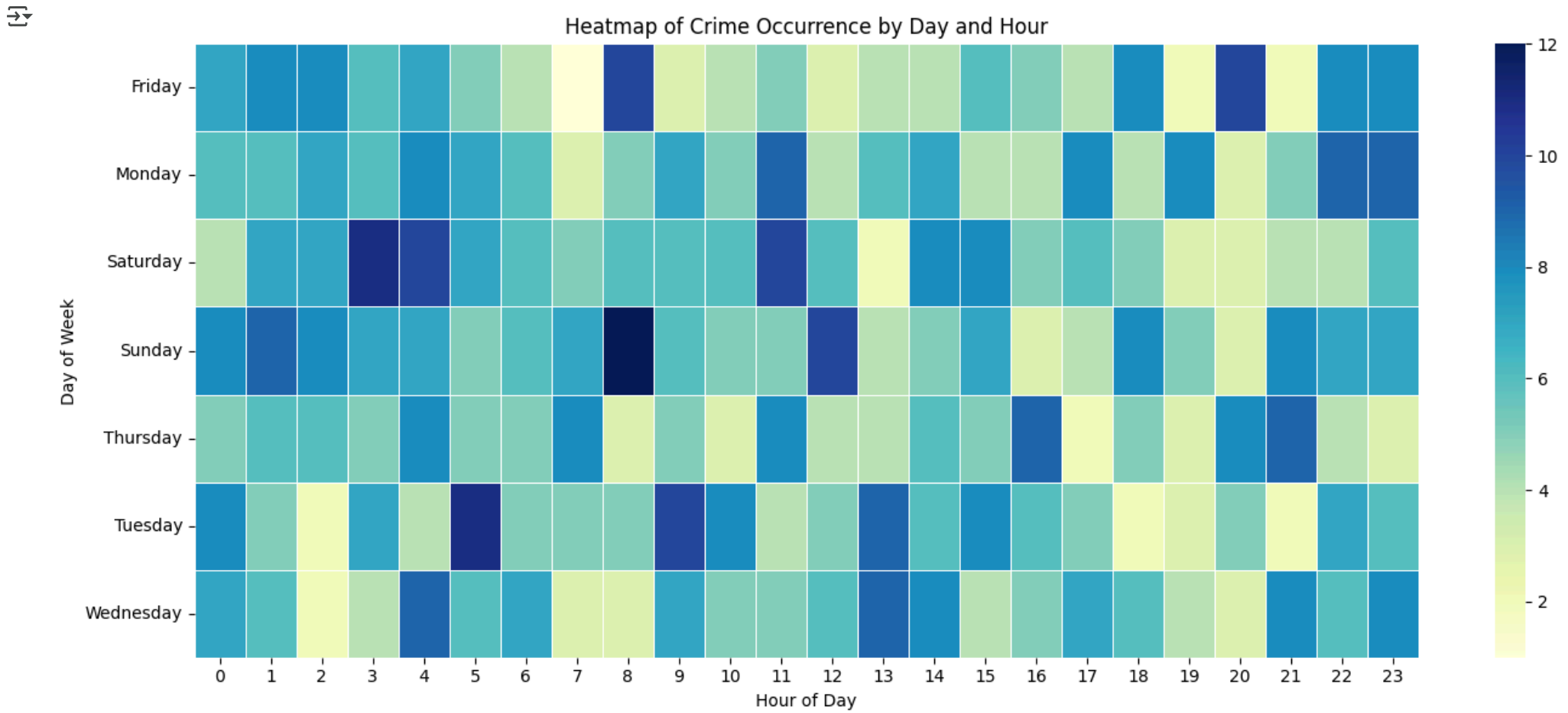
```
top_crimes = tn_df['Crime Description'].value_counts().nlargest(10)

plt.figure(figsize=(12, 6))
```

```
top_crimes.plot(kind='barh', color='skyblue')
plt.title('Top 10 Crime Descriptions in Tamil Nadu')
plt.xlabel('Number of Cases')
plt.ylabel('Crime Description')
plt.tight_layout()
plt.savefig('top_crime_types.png')
plt.show()
```




```
heatmap_data = tn_df.pivot_table(index='Day of Week', columns='Hour of Day', values='Crime Description', aggfunc='count')
plt.figure(figsize=(14, 6))
sns.heatmap(heatmap_data, cmap='YlGnBu', linewidths=.5)
plt.title('Heatmap of Crime Occurrence by Day and Hour')
plt.xlabel('Hour of Day')
```

```
plt.ylabel('Day of Week')
plt.tight_layout()
plt.savefig('crime_heatmap_day_hour.png')
plt.show()
```



** Provide critical insights and control suggestions.**

```
print(f"Total reported crimes: {len(tn_df)}")

most_common_domain = tn_df['Crime Domain'].value_counts().idxmax()
print(f"Most common crime domain: {most_common_domain}")

avg_age = tn_df['Victim Age'].mean()
```

```
print(f"Average victim age: {avg_age:.1f} years")

closed = tn_df['Case Closed'].value_counts(normalize=True).get('Yes', 0) * 100
print(f"Case closure rate: {closed:.2f}%")
```

```
Total reported crimes: 2493
Most common crime domain: Other Crime
Average victim age: 44.1 years
Case closure rate: 49.82%
```

```
print("- Deploy more patrols during peak hours identified in heatmap.")
print("- Focus on top 3 crime types for targeted interventions.")
print("- Launch awareness drives in high-crime age groups.")
print("- Invest in digital tools to improve case closure speed.")
```

```
- Deploy more patrols during peak hours identified in heatmap.
- Focus on top 3 crime types for targeted interventions.
- Launch awareness drives in high-crime age groups.
- Invest in digital tools to improve case closure speed.
```

Identify variables that correlate with crime trends

```
from scipy.stats import pearsonr

valid = tn_df.dropna(subset=['Victim Age', 'Police Deployed'])

age_police_corr, _ = pearsonr(valid['Victim Age'], valid['Police Deployed'])

print(f"🔗 Correlation between Victim Age and Police Deployed: {age_police_corr:.2f}")
print(f"🔗 Correlation between Police Deployed and Days to Close Case: {police_days_corr:.2f}")
```

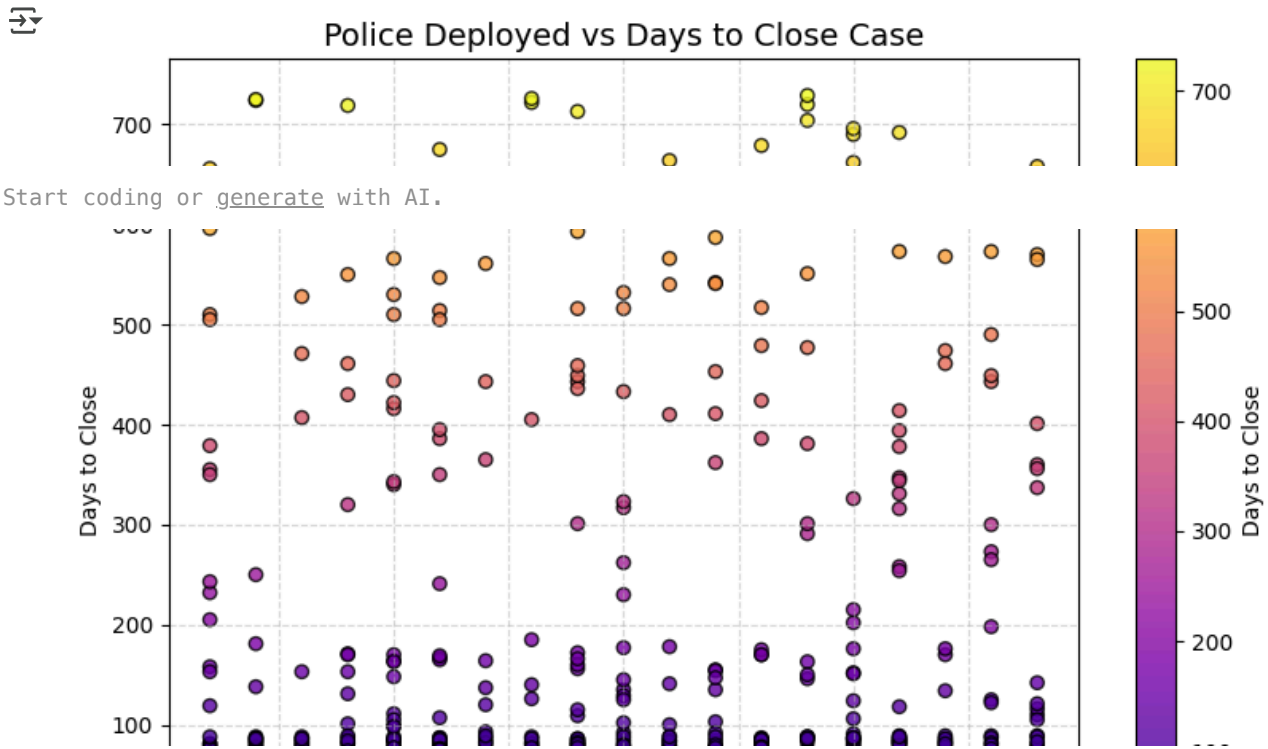
```
🔗 Correlation between Victim Age and Police Deployed: -0.04
🔗 Correlation between Police Deployed and Days to Close Case: 0.02
```

```
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))

plt.scatter(valid['Victim Age'], valid['Police Deployed'],
            color='#ff0040', alpha=0.7, s=60, label='Victim Age')
```

```
    plt.scatter(valid['Victim Age'], valid['Police Deployed'],
                color='#4000ff', alpha=0.3, s=60, label='Police Deployed')

    plt.title('Victim Age vs Police Deployed', fontsize=16)
    plt.xlabel('Victim Age', fontsize=12)
    plt.ylabel('Police Deployed', fontsize=12)
    plt.grid(True, linestyle='--', alpha=0.5)
    plt.legend()
    plt.tight_layout()
    plt.savefig('victim_age_vs_police_bright.png')
    plt.show()
```


```
    import matplotlib.pyplot as plt
```

```
plt.figure(figsize=(8, 6))

sc = plt.scatter(
    valid_cases['Police Deployed'],
    valid_cases['Days to Close'],
    c=valid_cases['Days to Close'],
    cmap='plasma',
    alpha=0.8,
    edgecolor='black'
)

plt.colorbar(sc, label='Days to Close')
plt.title('Police Deployed vs Days to Close Case', fontsize=14)
plt.xlabel('Police Deployed')
plt.ylabel('Days to Close')
plt.grid(True, linestyle='--', alpha=0.5)
plt.tight_layout()
plt.savefig('colored_police_vs_days_scatter.png')
plt.show()
```



Start coding or generate with AI.