Crime Data Exploration

Mini Capstone Project: Crime Data Analysis With Mysql And Python

Contents

- Project Overview
- Spatial Analysis
- Victim Demographics
- Location Analysis
- Crime Code Analysis
- Tools and Libraries
- Tools and Libraries
- Insights

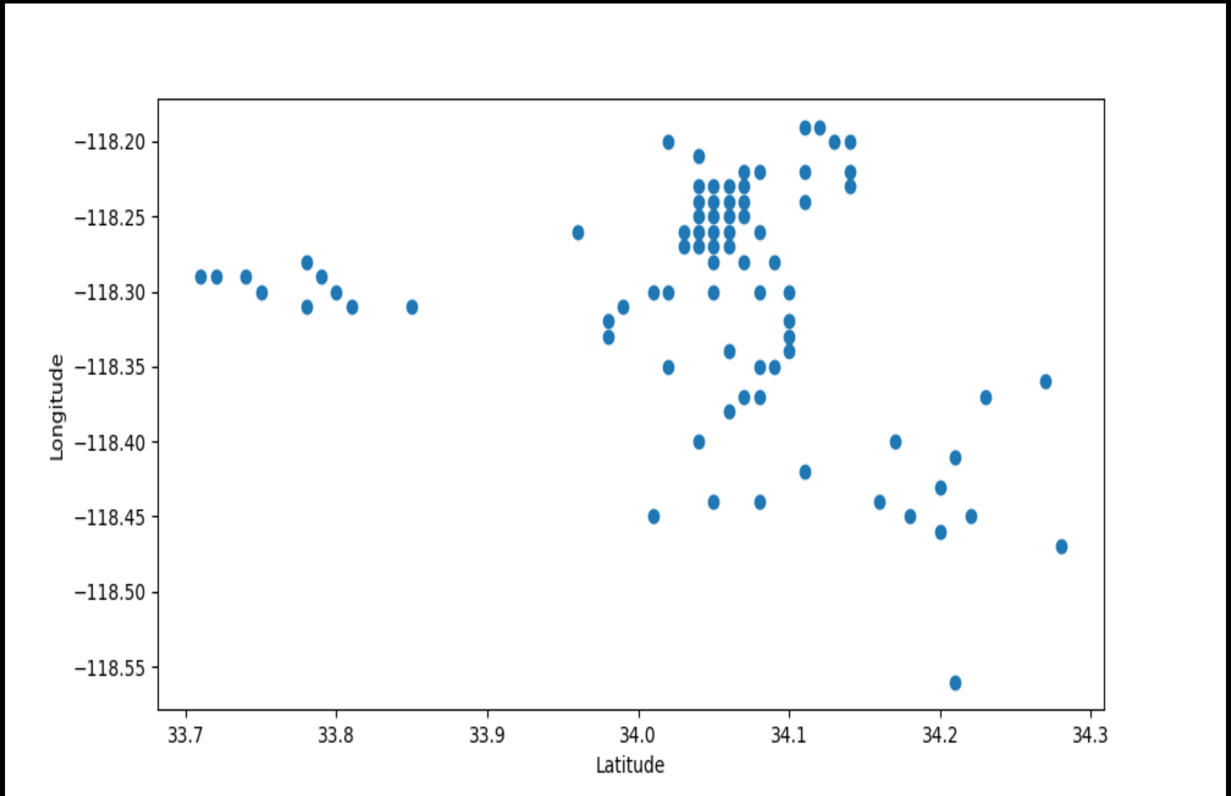
Project Overview

We're diving deep into crime data to understand it better. By analyzing spatial patterns, victim demographics, location-specific occurrences, and crime code data, we aim to unveil hidden trends, correlations, and hotspots within the dataset.

By finding these patterns, we can help make neighbourhood safer by focusing on the right areas and issues. Our goal is to use facts and numbers to guide decisions and actions, making our communities safer and reducing crime.

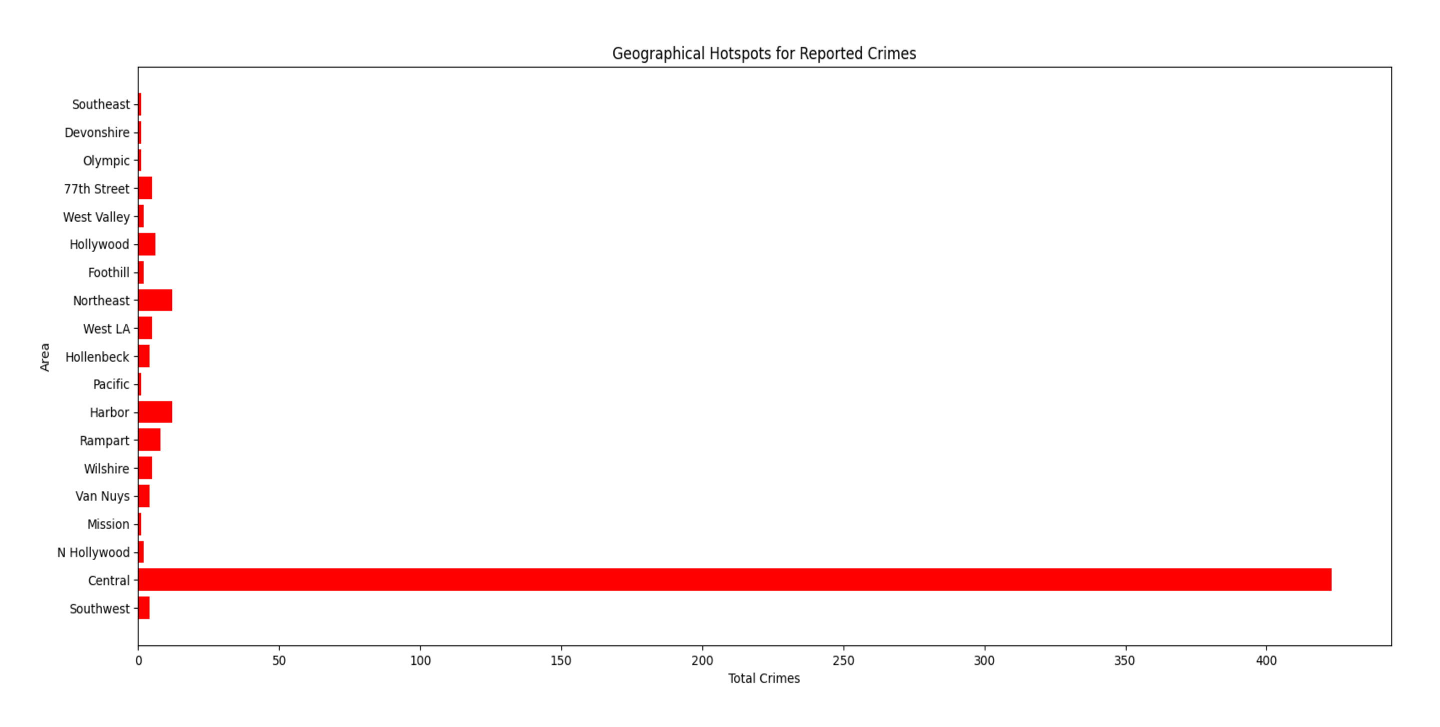
Spatial Analysis

In our spatial analysis, we pinpoint geographical hotspots for reported crimes using a scatterplot. Focusing on latitude and longitude, we'll map crime incidents. This helps identify areas with high crime rates, aiding in targeted law enforcement and crime prevention strategies.



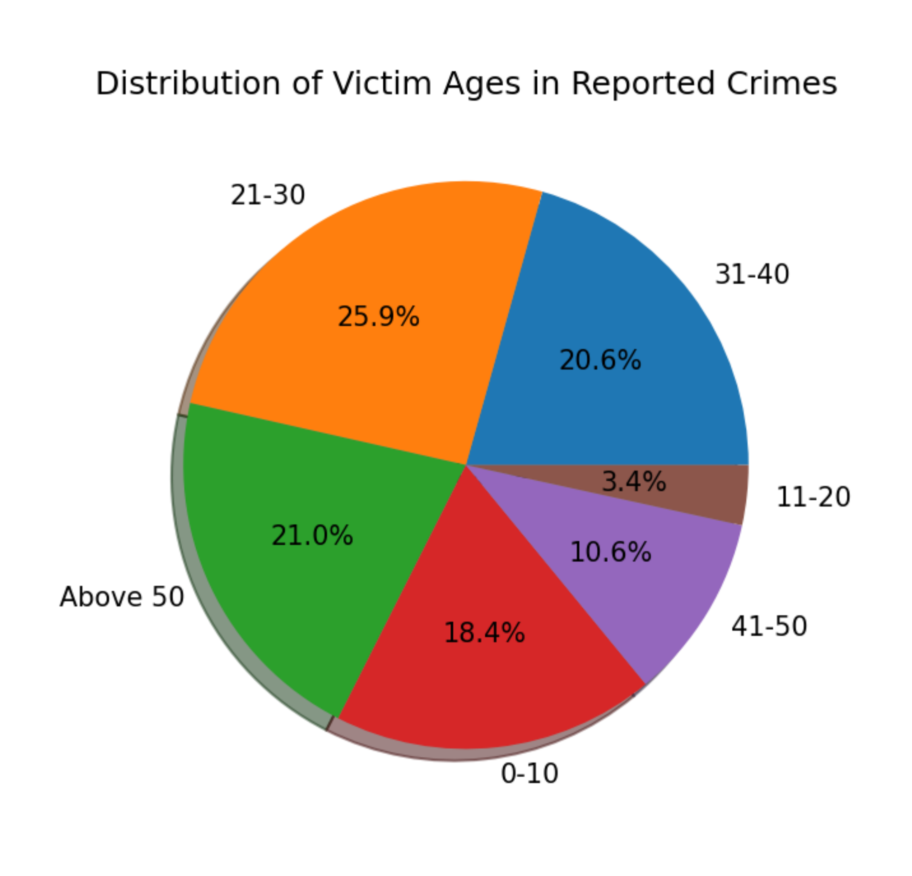
Spatial Analysis

In another spatial analysis, we employ a bar graph to pinpoint specific hotspots of reported crimes. By analyzing crime frequency across different areas, we identify and highlight specific locations with the highest crime rates. This targeted approach aids in allocating resources effectively for crime mitigation strategies.



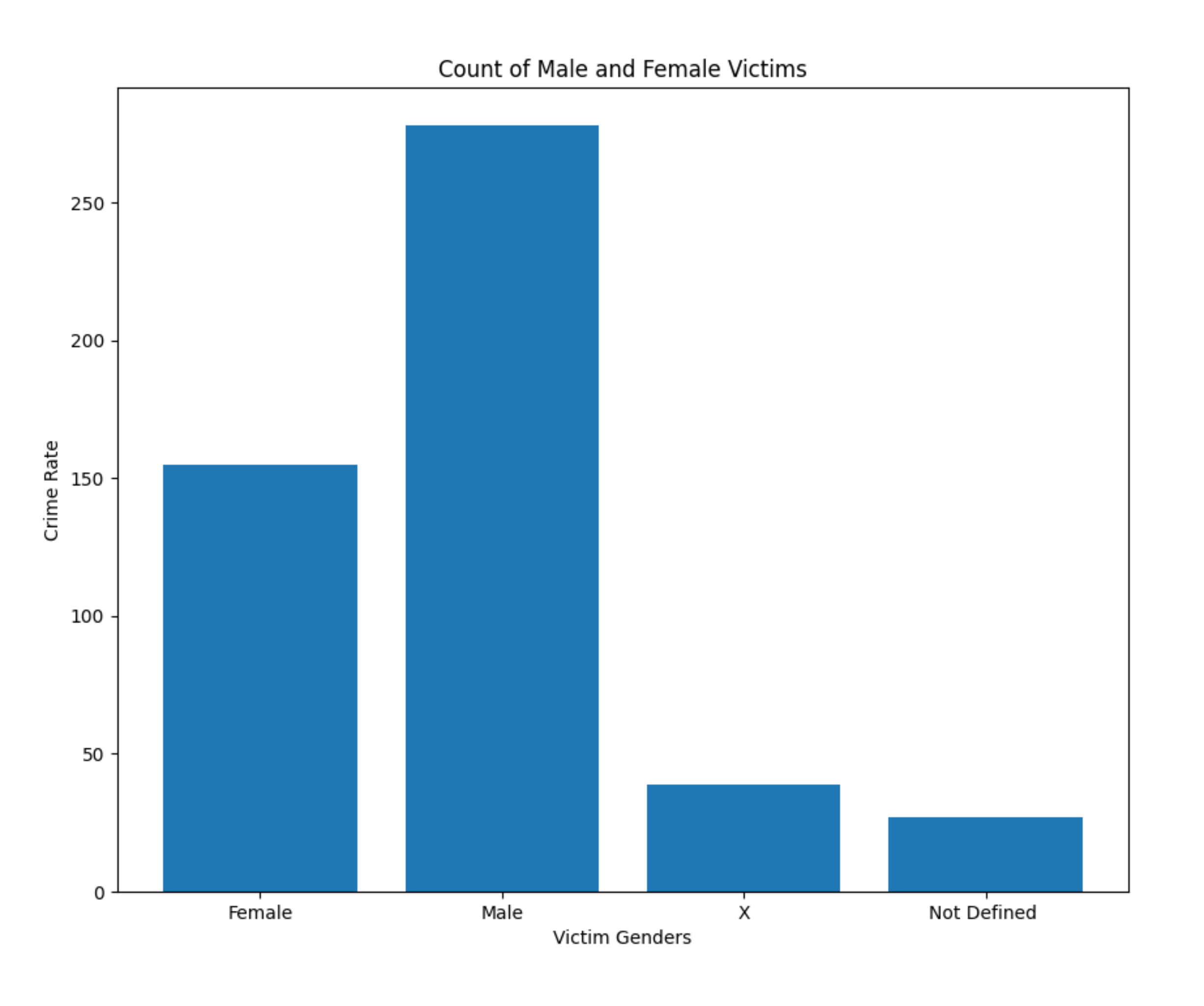
Victim Demographics

In Victim Demographics, we examine the distribution of victim ages in reported crimes using a pie chart. Predominantly, victims fall within the 21 to 30 age group, indicating a significant proportion of young individuals affected. This insight informs targeted interventions to address vulnerabilities within this demographic.



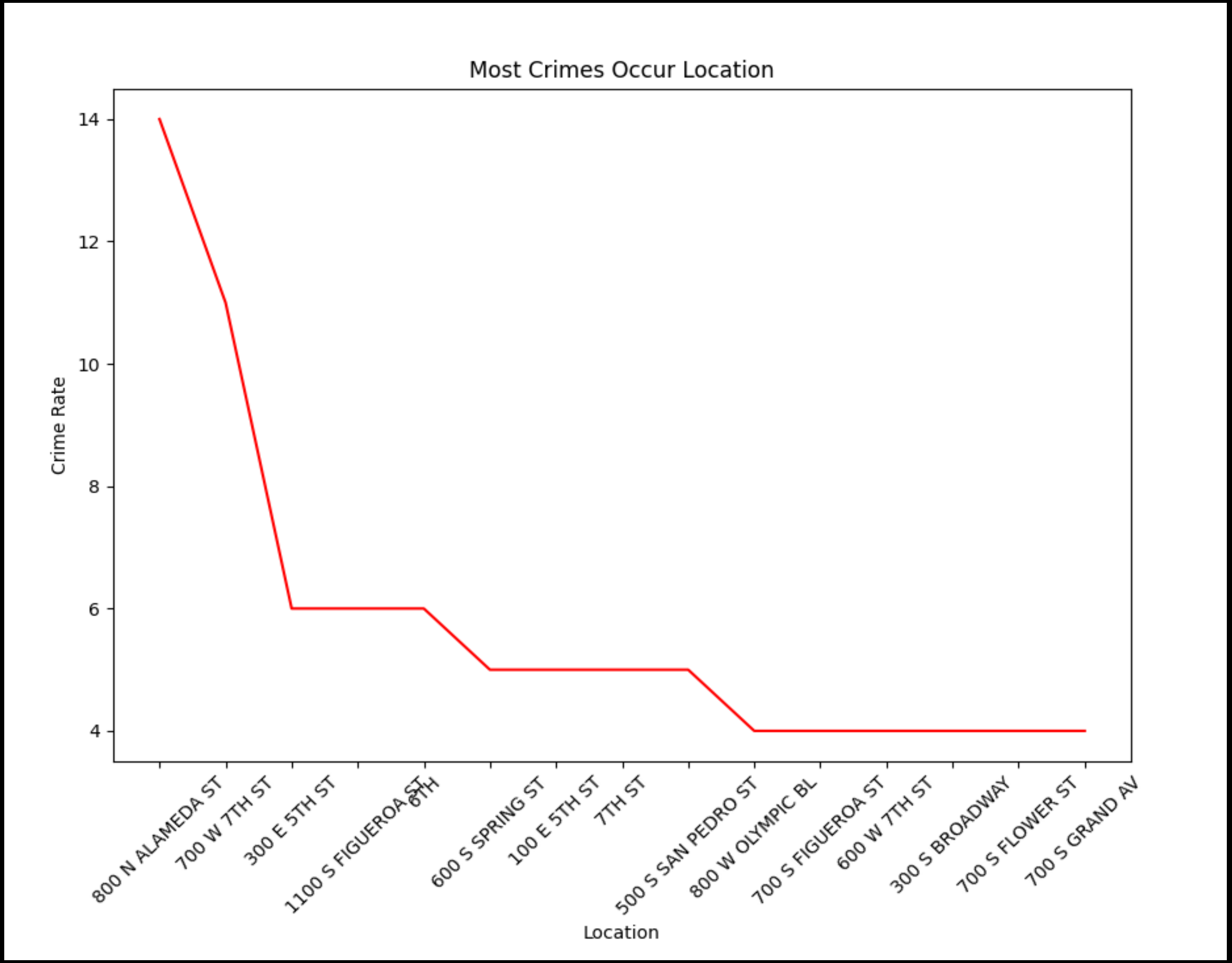
Victim Demographics

In another study of victim demographics, we used a bar graph to find difference in crime rates between males and females. The data showed that males experience more crimes, making up the majority of victims. This highlights the importance of gender-aware crime prevention efforts to tackle this issue effectively.



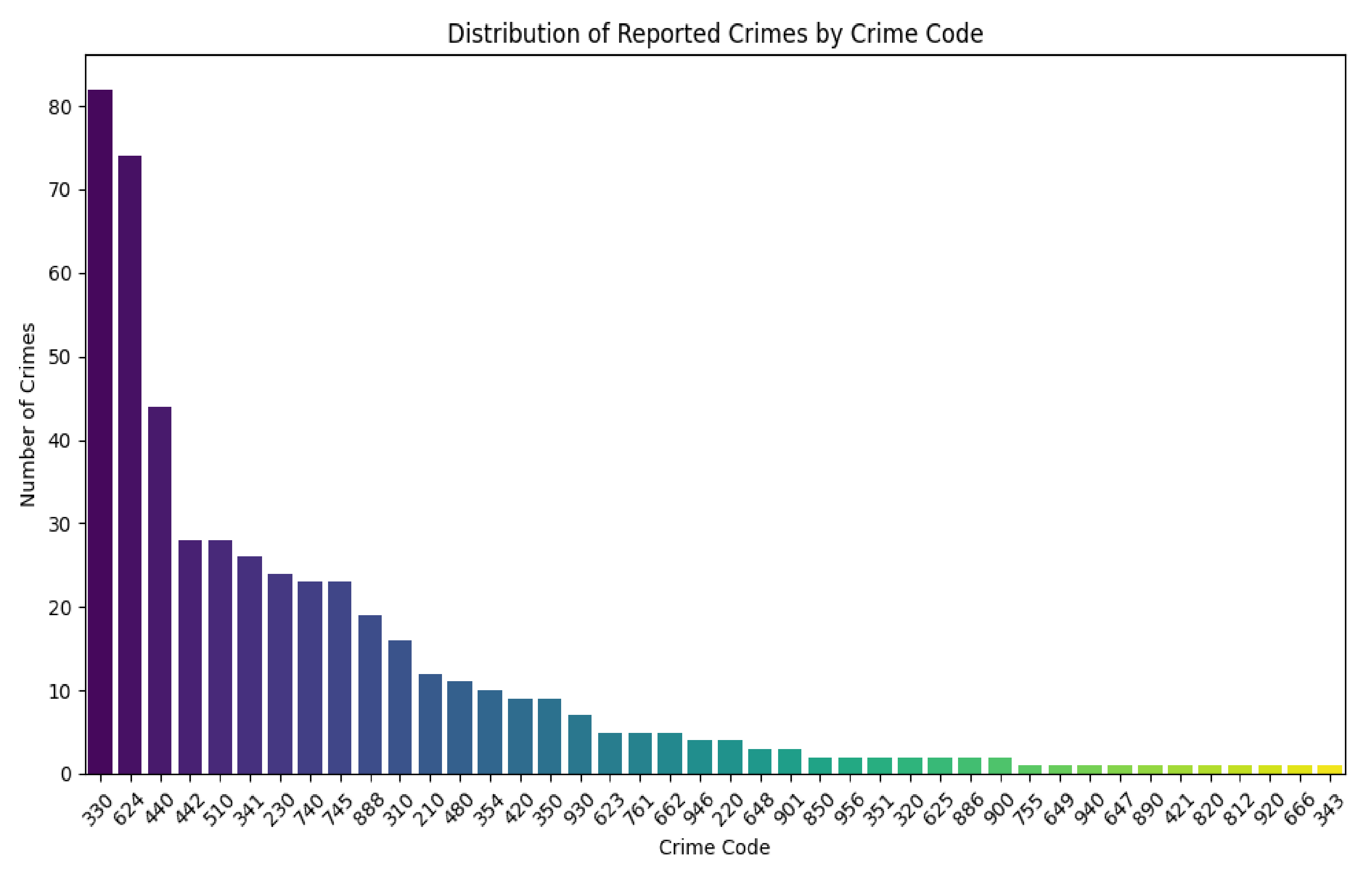
Location Analysis

In Location Analysis, we identify locations with the highest crime rates. Using a line graph, we pinpoint these crime hotspots. Here, we can see ‘800 N Alameda ST’ is the prime location of crimes. This data-driven approach helps focus law enforcement resources on key areas, improving crime management and public safety strategies.



Crime Code Analysis

In Crime Code Analysis, we analyze the distribution of reported crimes based on Crime Code using a count plot. This visualization reveals the types of crimes and their respective frequencies by code. Understanding these patterns aids in developing targeted crime prevention measures and allocating resources more effectively.


Distribution of Reported Crimes by Crime Code

Tools And Libraries

I used Visual Studio Code (VS Code) for writing Python code. It’s great because it has lots of useful features and is easy to use. The debugging tools and version control features helped me a lot, and I could customize it to suit my needs, making coding more efficient and enjoyable.

PyMySQL

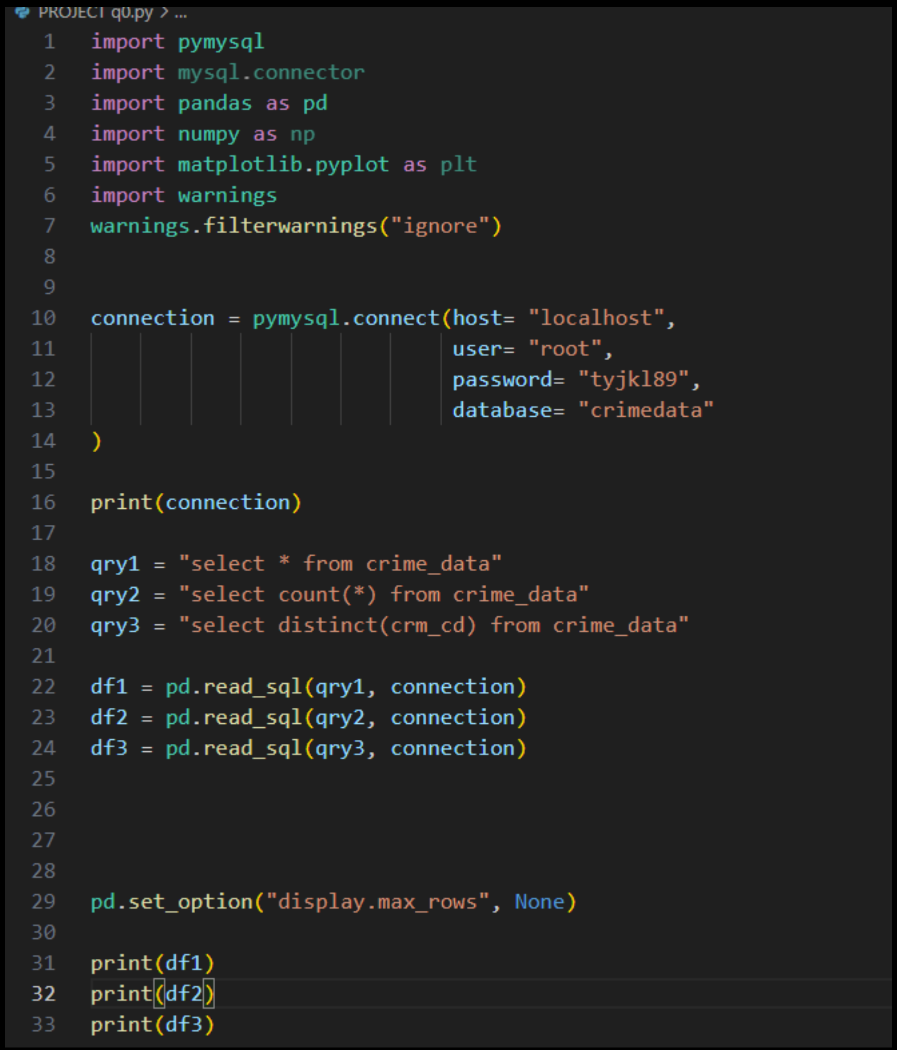
The PyMySQL library in Python enables seamless interaction with MySQL databases, allowing you to execute SQL queries, retrieve data, and perform database operations directly from your Python code. It simplifies database connectivity tasks, making it easier to work with MySQL databases in various Python applications such as data analysis, web development, and automation scripts.

Matplotlib and Seaborn

Matplotlib and Seaborn are powerful Python libraries for data visualization. Matplotlib offers a wide range of customizable plots like line charts, histograms, and scatter plots, ideal for exploring data and presenting insights. Seaborn, built on top of Matplotlib, provides additional functionalities and beautiful default styles for creating appealing statistical graphics, making it a favorite for data analysts and researchers worldwide.

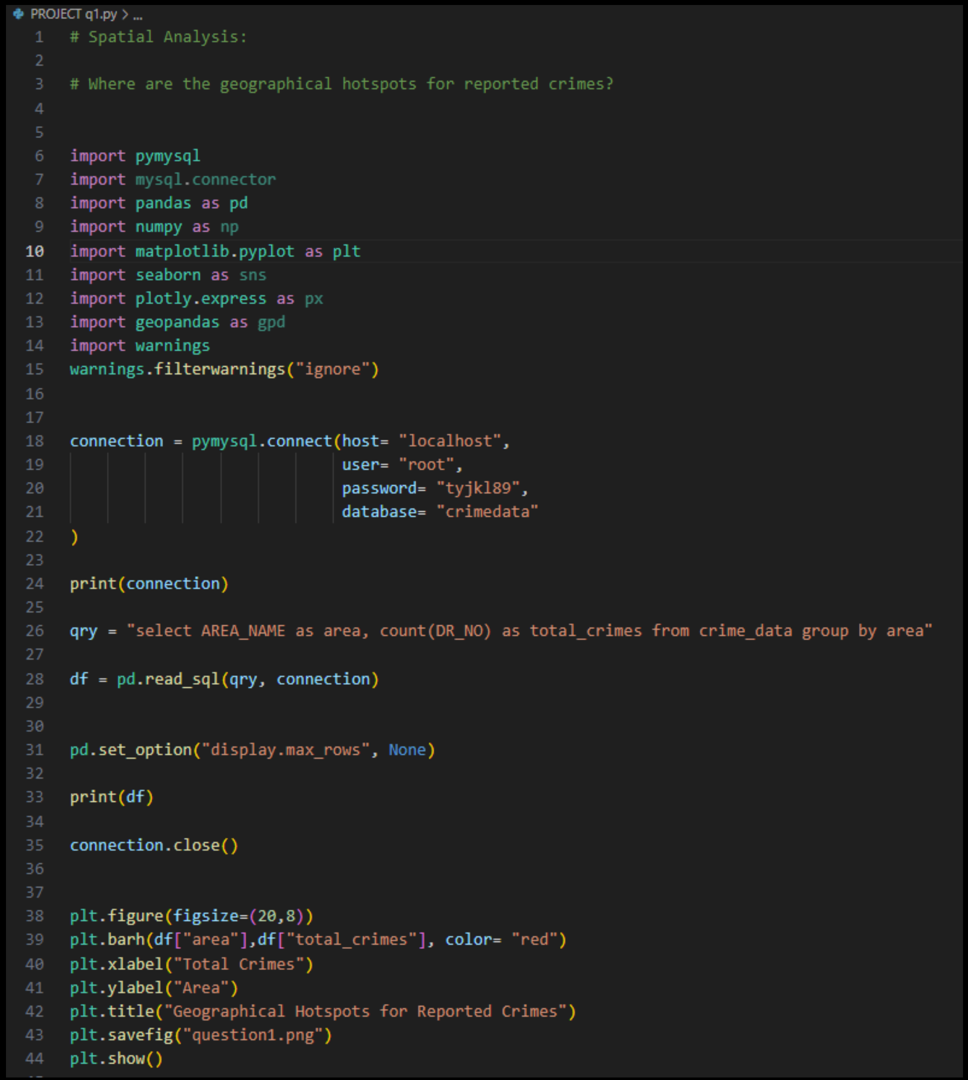
Python Scripts

database setup, data import



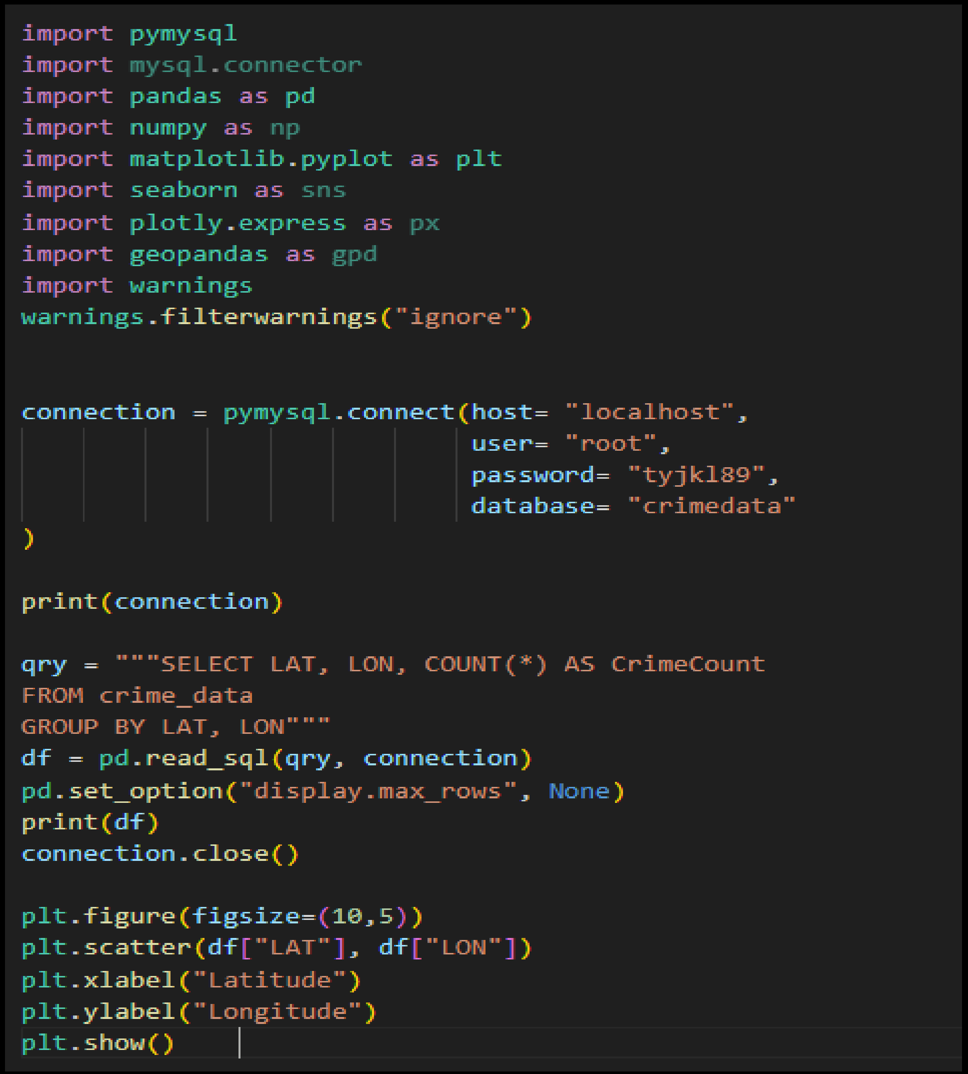
```python
import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")


connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjkl89",
                             database= "crimedata"
)

print(connection)

qry1 = "select * from crime_data"
qry2 = "select count(*) from crime_data"
qry3 = "select distinct(crm_cd) from crime_data"

df1 = pd.read_sql(qry1, connection)
df2 = pd.read_sql(qry2, connection)
df3 = pd.read_sql(qry3, connection)




pd.set_option("display.max_rows", None)

print(df1)
print(df2)
print(df3)
```

Spatial Analysis (bar graph)



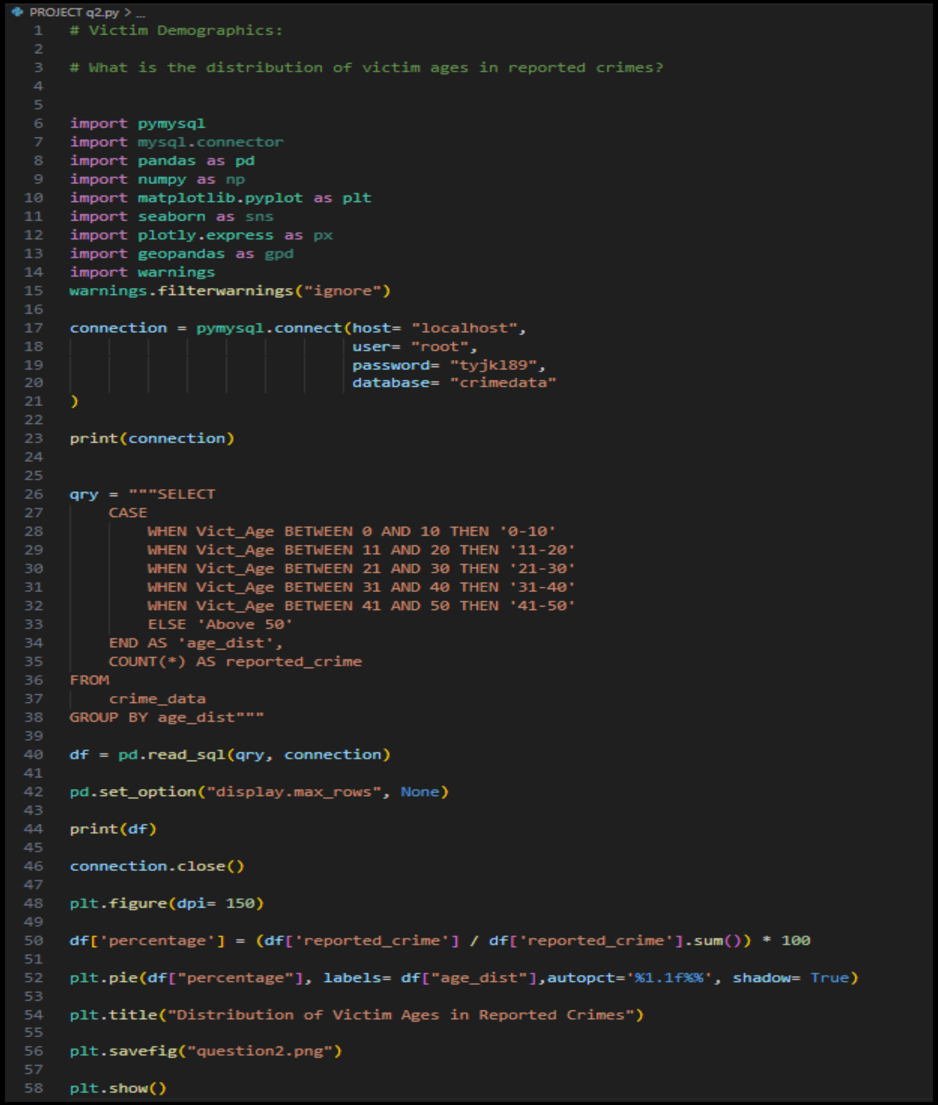
```
# Spatial Analysis:

# Where are the geographical hotspots for reported crimes?


import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")


connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjkl89",
                             database= "crimedata"
)

print(connection)

qry = "select AREA_NAME as area, count(DR_NO) as total_crimes from crime_data group by area"

df = pd.read_sql(qry, connection)


pd.set_option("display.max_rows", None)

print(df)

connection.close()


plt.figure(figsize=(20,8))
plt.barh(df["area"],df["total_crimes"], color= "red")
plt.xlabel("Total Crimes")
plt.ylabel("Area")
plt.title("Geographical Hotspots for Reported Crimes")
plt.savefig("question1.png")
plt.show()
```

Spatial Analysis (scatterplot)



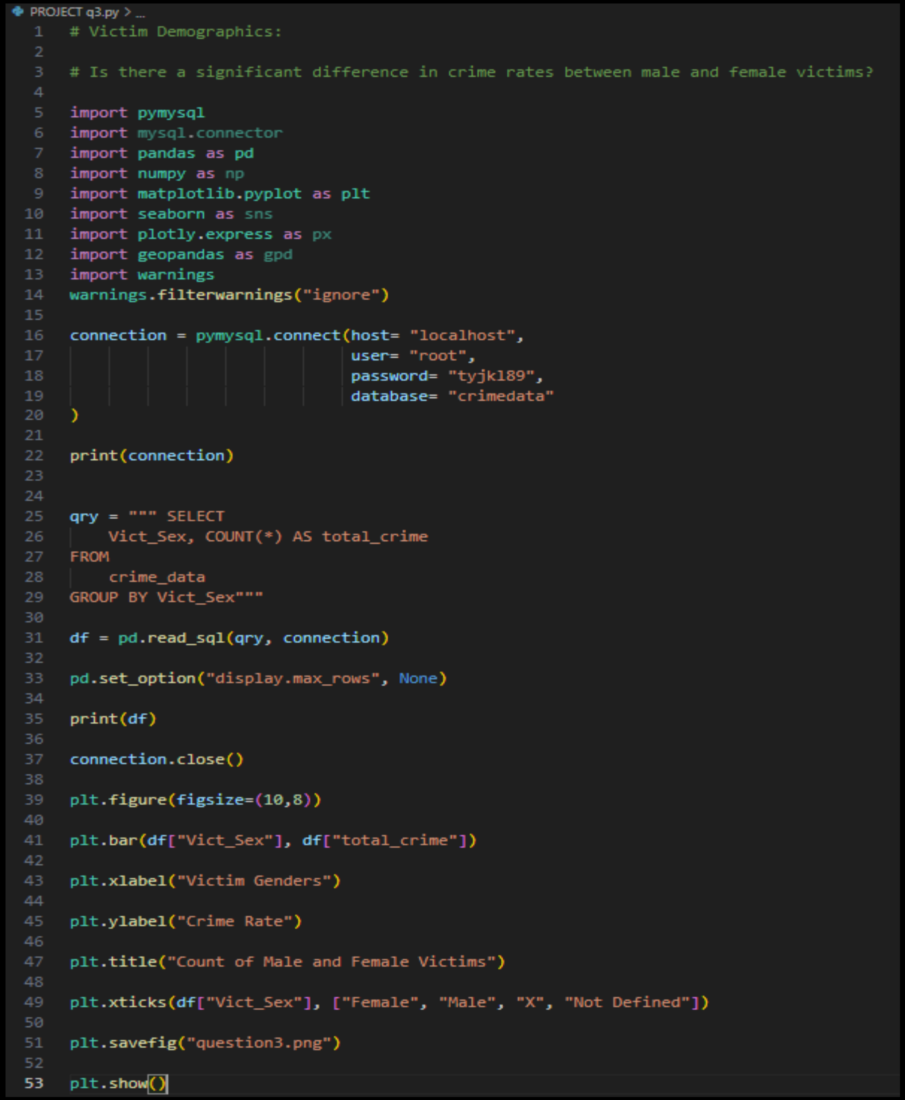
```
import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")


connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjkl89",
                             database= "crimedata"
)

print(connection)

qry = """SELECT LAT, LON, COUNT(*) AS CrimeCount
FROM crime_data
GROUP BY LAT, LON"""
df = pd.read_sql(qry, connection)
pd.set_option("display.max_rows", None)
print(df)
connection.close()

plt.figure(figsize=(10,5))
plt.scatter(df["LAT"], df["LON"])
plt.xlabel("Latitude")
plt.ylabel("Longitude")
plt.show()
```

Victim Demographics (pie chart)



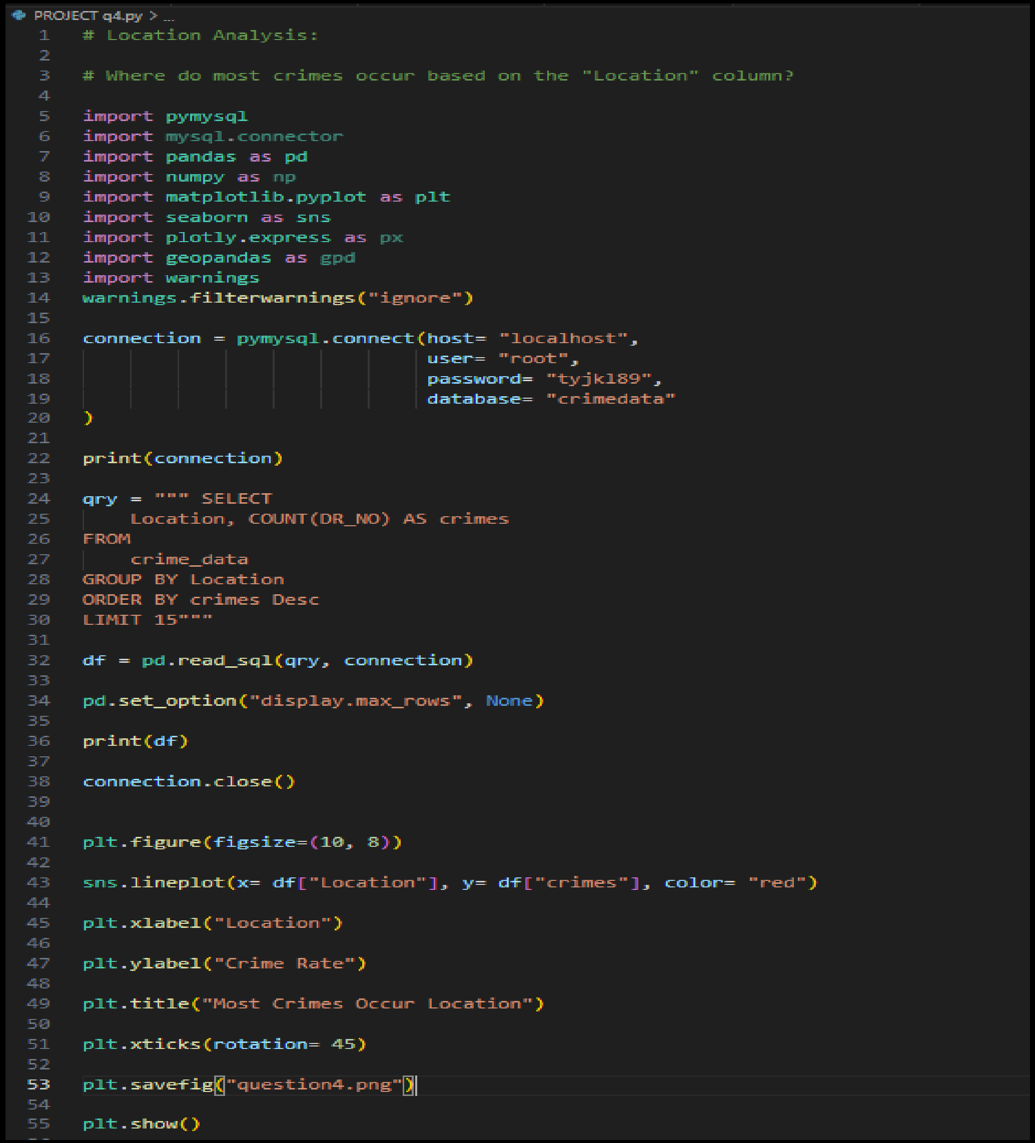
```python
# Victim Demographics:

# What is the distribution of victim ages in reported crimes?


import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")

connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjk189",
                             database= "crimedata"
)

print(connection)


qry = """SELECT
    CASE
        WHEN Vict_Age BETWEEN 0 AND 10 THEN '0-10'
        WHEN Vict_Age BETWEEN 11 AND 20 THEN '11-20'
        WHEN Vict_Age BETWEEN 21 AND 30 THEN '21-30'
        WHEN Vict_Age BETWEEN 31 AND 40 THEN '31-40'
        WHEN Vict_Age BETWEEN 41 AND 50 THEN '41-50'
        ELSE 'Above 50'
    END AS 'age_dist',
    COUNT(*) AS reported_crime
FROM
    crime_data
GROUP BY age_dist"""

df = pd.read_sql(qry, connection)

pd.set_option("display.max_rows", None)

print(df)

connection.close()

plt.figure(dpi= 150)

df['percentage'] = (df['reported_crime'] / df['reported_crime'].sum()) * 100

plt.pie(df["percentage"], labels= df["age_dist"],autopct='%1.1f%%', shadow= True)

plt.title("Distribution of Victim Ages in Reported Crimes")

plt.savefig("question2.png")

plt.show()
```

Victim Demographics (bar graph)



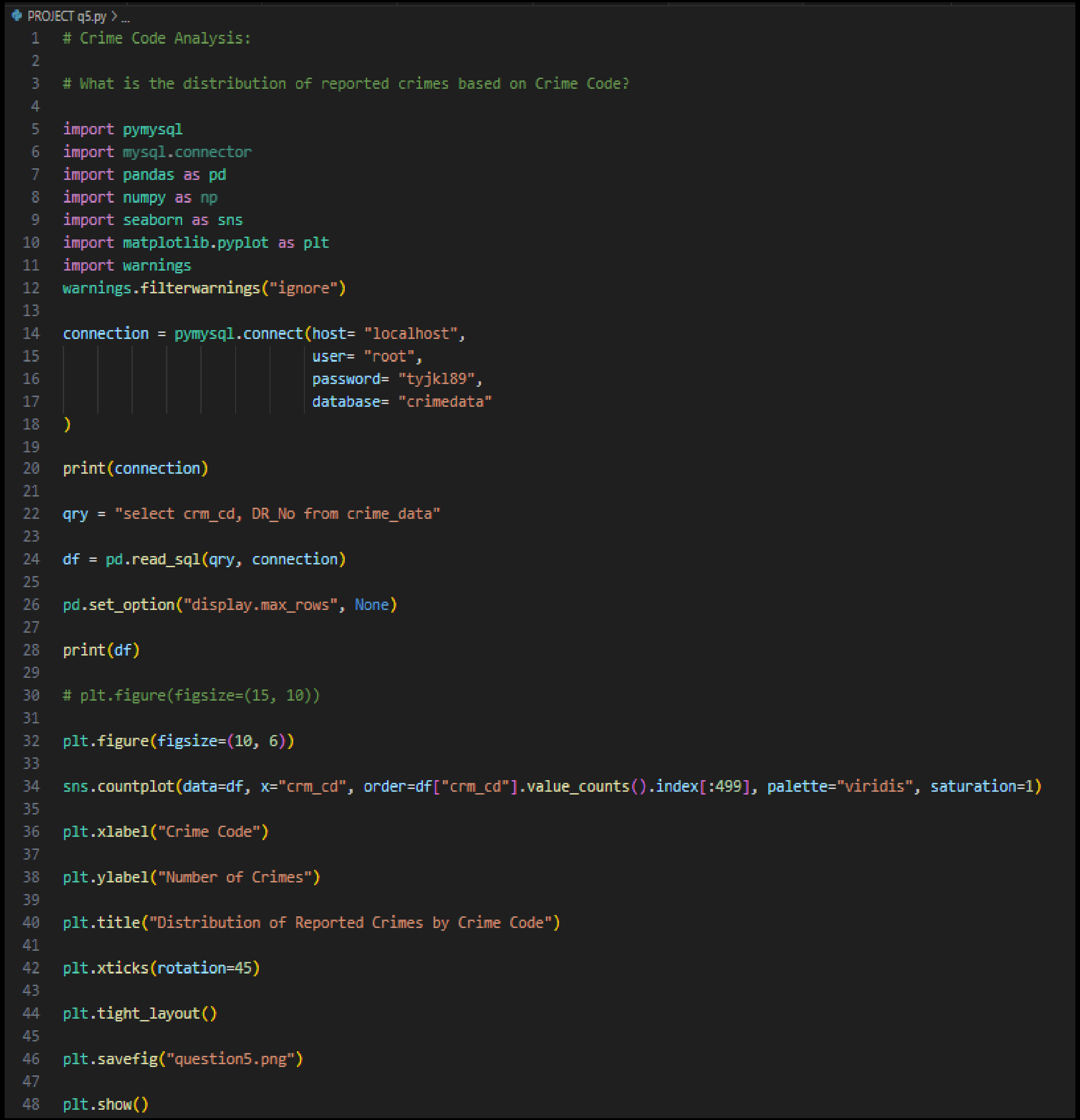
```python
# Victim Demographics:

# Is there a significant difference in crime rates between male and female victims?

import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")

connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjk189",
                             database= "crimedata"
)

print(connection)


qry = """ SELECT
    Vict_Sex, COUNT(*) AS total_crime
FROM
    crime_data
GROUP BY Vict_Sex"""

df = pd.read_sql(qry, connection)

pd.set_option("display.max_rows", None)

print(df)

connection.close()

plt.figure(figsize=(10,8))

plt.bar(df["Vict_Sex"], df["total_crime"])

plt.xlabel("Victim Genders")

plt.ylabel("Crime Rate")

plt.title("Count of Male and Female Victims")

plt.xticks(df["Vict_Sex"], ["Female", "Male", "X", "Not Defined"])

plt.savefig("question3.png")

plt.show()
```

Location Analysis



```python
# Location Analysis:

# Where do most crimes occur based on the "Location" column?

import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")

connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjkl89",
                             database= "crimedata"
)

print(connection)

qry = """ SELECT
    Location, COUNT(DR_NO) AS crimes
FROM
    crime_data
GROUP BY Location
ORDER BY crimes Desc
LIMIT 15"""

df = pd.read_sql(qry, connection)

pd.set_option("display.max_rows", None)

print(df)

connection.close()


plt.figure(figsize=(10, 8))

sns.lineplot(x= df["Location"], y= df["crimes"], color= "red")

plt.xlabel("Location")

plt.ylabel("Crime Rate")

plt.title("Most Crimes Occur Location")

plt.xticks(rotation= 45)

plt.savefig("question4.png")

plt.show()
```

Crime Code Analysis



```python
# Crime Code Analysis:

# What is the distribution of reported crimes based on Crime Code?

import pymysql
import mysql.connector
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")

connection = pymysql.connect(host= "localhost",
                             user= "root",
                             password= "tyjkl89",
                             database= "crimedata"
)

print(connection)

qry = "select crm_cd, DR_No from crime_data"

df = pd.read_sql(qry, connection)

pd.set_option("display.max_rows", None)

print(df)

# plt.figure(figsize=(15, 10))

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

sns.countplot(data=df, x="crm_cd", order=df["crm_cd"].value_counts().index[:499], palette="viridis", saturation=1)

plt.xlabel("Crime Code")

plt.ylabel("Number of Crimes")

plt.title("Distribution of Reported Crimes by Crime Code")

plt.xticks(rotation=45)

plt.tight_layout()

plt.savefig("question5.png")

plt.show()
```

Insights

Based on our analysis, we recommend that law enforcement concentrate their efforts on areas where crimes occur most frequently. We also suggest allocating resources based on the ages of most victims and the types of crimes common in different areas. This approach can help in better crime prevention and control.

THANK
YOU