Practical No-8

Date of Conduction :   **Date of Checking:**

Data Visualization I

1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains information about the passengers who boarded the unfortunate Titanic ship. Use the Seaborn library to see if we can find any patterns in the data.

2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram.

Python Code

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

# Load the Titanic dataset
titanic = sns.load_dataset('titanic')

# 1. Visualizing patterns in the data
sns.set(style="whitegrid")
plt.figure(figsize=(12, 6))

# Use Seaborn's countplot to visualize the number of passengers in each
class
sns.countplot(x='class', hue='survived', data=titanic, palette='Set1')
plt.title('Survival Count by Passenger Class')
plt.show()

# 2. Plotting a histogram for the distribution of ticket prices (fare)
plt.figure(figsize=(12, 6))

# Use Seaborn's histogram to visualize the distribution of ticket prices
sns.histplot(titanic['fare'], bins=30, kde=True, color='skyblue')
plt.title('Distribution of Ticket Prices (Fare)')
plt.xlabel('Fare')
plt.ylabel('Frequency')
plt.show()
```

Explanation:

Visualizing patterns in the data:

- This part uses Seaborn's countplot to visualize the number of passengers in each class (1st, 2nd, 3rd) and their survival status. The color hue represents survival (survived or not survived).
- Plotting a histogram for ticket prices (fare):

- This part uses Seaborn's histplot to plot a histogram for the distribution of ticket prices (fare column) with 30 bins and a kernel density estimate (kde) for a smooth representation of the distribution.

- Make sure to have Seaborn and Matplotlib installed (pip install seaborn matplotlib) before running the code. You can adjust the parameters as needed based on your preferences and analysis goals.

OUTPUT:

Survival Count by Passenger Class

Distribution of Ticket Prices (Fare)