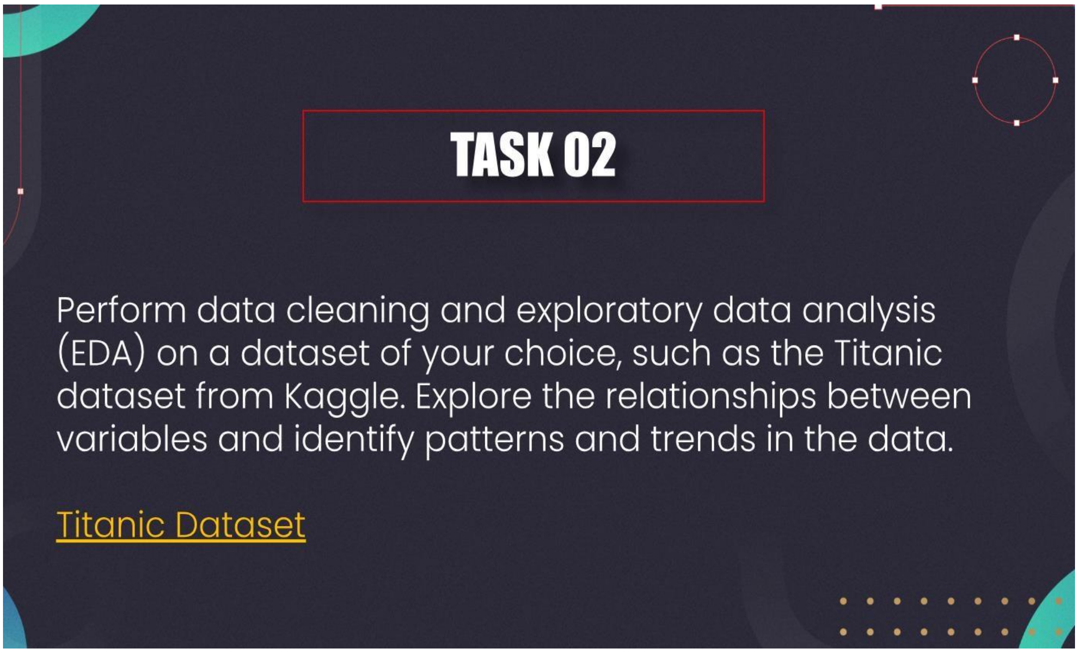
Project Name: Analyzing Titanic Passenger Data: Data Cleaning and Exploratory Data Analysis

By Shruti Thorat

Project Introduction

- The Titanic disaster is one of the most infamous shipwrecks in history.
- On April 15, 1912, the Titanic sank after colliding with an iceberg, resulting in the deaths of more than 1,500 passengers and crew.
- This project aims to analyze the passenger data from the Titanic to uncover insights into the factors that influenced survival rates.
- By performing data cleaning and exploratory data analysis (EDA), we will explore relationships between variables and identify patterns and trends in the data.



Project summary

This project involves the following steps:

1. **Data Cleaning**: Handling missing values, removing duplicates, and transforming data to ensure accuracy and consistency.
2. **Exploratory Data Analysis (EDA)**: Visualizing data to uncover relationships between different variables and identify significant patterns.
3. **Insights and Trends**: Analyzing the cleaned data to draw meaningful conclusions about the factors affecting passenger survival rates on the Titanic.

Business Objective

 The primary objective of this project is to gain a deeper understanding of the factors that influenced the survival rates of passengers on the Titanic. By analyzing the dataset, we aim to:

1. Identify key variables that had a significant impact on survival rates, such as passenger class, age, gender, fare, and embarked port.
2. Provide visualizations that clearly depict these relationships and trends.
3. Offer insights that can inform future safety measures and decision-making processes in maritime travel and disaster management.

By achieving these objectives, the project seeks to contribute valuable knowledge to the historical analysis of the Titanic disaster and enhance data-driven decision-making in related fields

Steps :-

Step 1: Importing Libraries

Step 2: Loading the Dataset

Step 3: Understanding the Data

Step 4: Handling Missing Values

Step 5: Data Cleaning

Step 6: Exploratory Data Analysis (EDA)

Step 1: Importing Libraries

```
[ ] #importing Libraries
    import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns
```

Step 2: Loading the Dataset

```
#Loding the Dataset
data=pd.read_csv('/content/Titanic.csv')
data
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

Step 3: Understanding the Data

```
[ ] data.head(7)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S

```
[ ] data.tail()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

```
[ ] data.shape
(891, 12)
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

```
[ ] data.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
[ ] data.columns
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

```
[ ] duplicated_value= data.duplicated().value_counts
    print(duplicated_value)
```

```
<bound method IndexOpsMixin.value_counts of 0      False
1      False
2      False
3      False
4      False
       ...  
886    False
887    False
888    False
889    False
890    False
Length: 891, dtype: bool>
```

```
[ ] data.duplicated().sum()
```

```
0
```

```
[ ] print(data.isnull().sum())
```

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

Step 4: Handling Missing Values

```
[ ] #Handling Missing Values
    data['Age'].fillna(data['Age'].median(),inplace=True)
```

```
<ipython-input-15-5cd2b3b925d2>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the original


  data['Age'].fillna(data['Age'].median(),inplace=True)
```

```
[ ] data.drop(columns=['Cabin'],inplace=True)
```

```
data['Embarked'].fillna(data['Embarked'].mode()[0],inplace=True)
```

```
<ipython-input-18-7cda4401dcd3>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the original


  data['Embarked'].fillna(data['Embarked'].mode()[0],inplace=True)
```

Step 5: Data Cleaning

```
#Data Cleaning
print(data.isnull().sum())
```

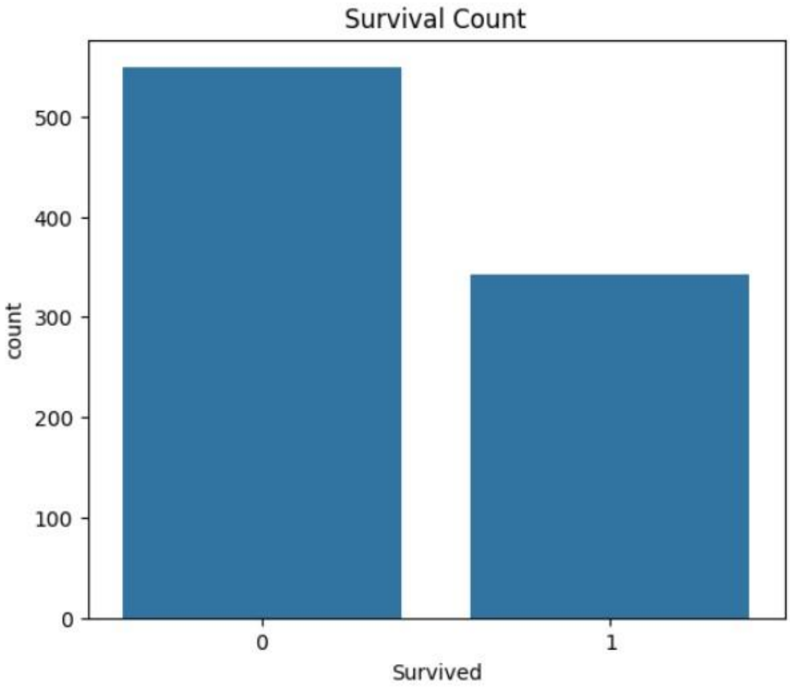
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

Step 6: Exploratory Data Analysis (EDA)

Perform EDA to explore relationships between variables and identify patterns and trends.

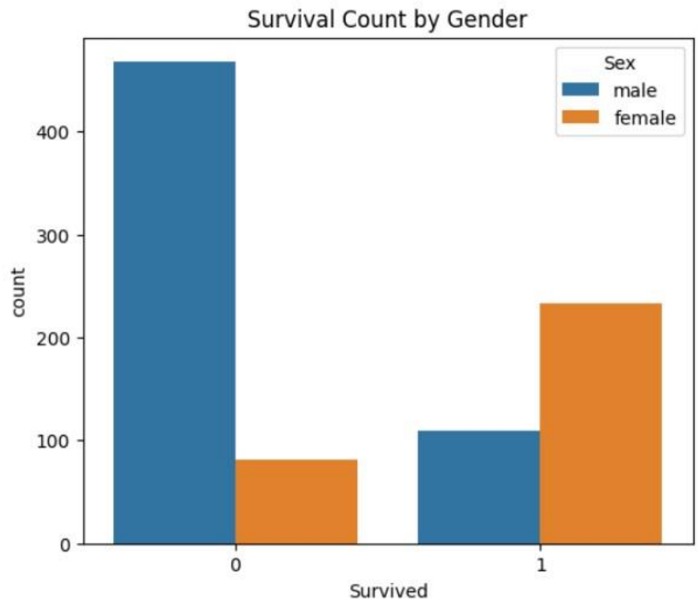
#Survived vs. Not Survived

```
[ ] #Exploratory Data Analysis(EDA)
    plt.figure(figsize=(6,5))
    sns.countplot(data=data,x='Survived')
    plt.title('Survival Count')
    plt.show()
```



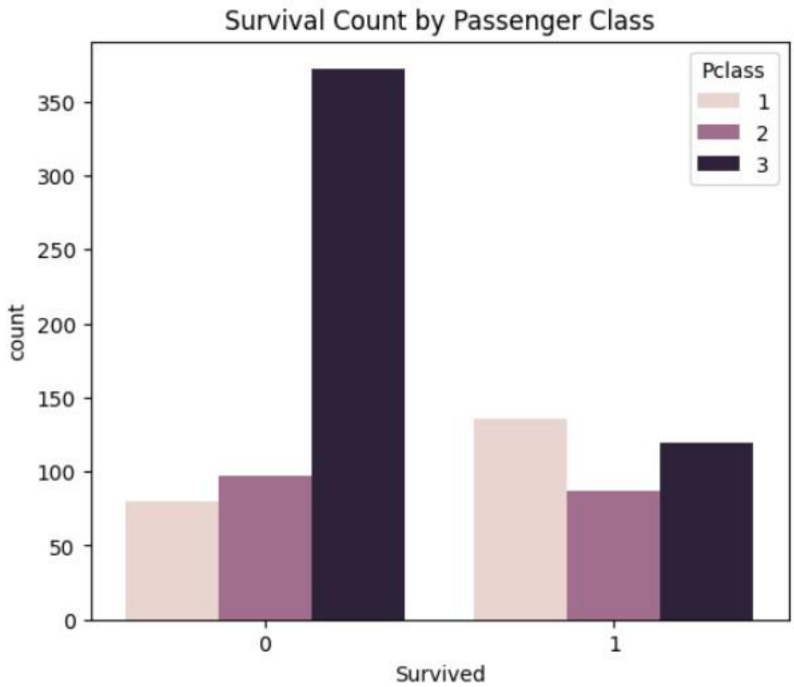
Survival Rate by Sex

```python
plt.figure(figsize=(6,5))
sns.countplot(data=data,x="Survived", hue="Sex")
plt.title("Survival Count by Gender")
plt.show()
```



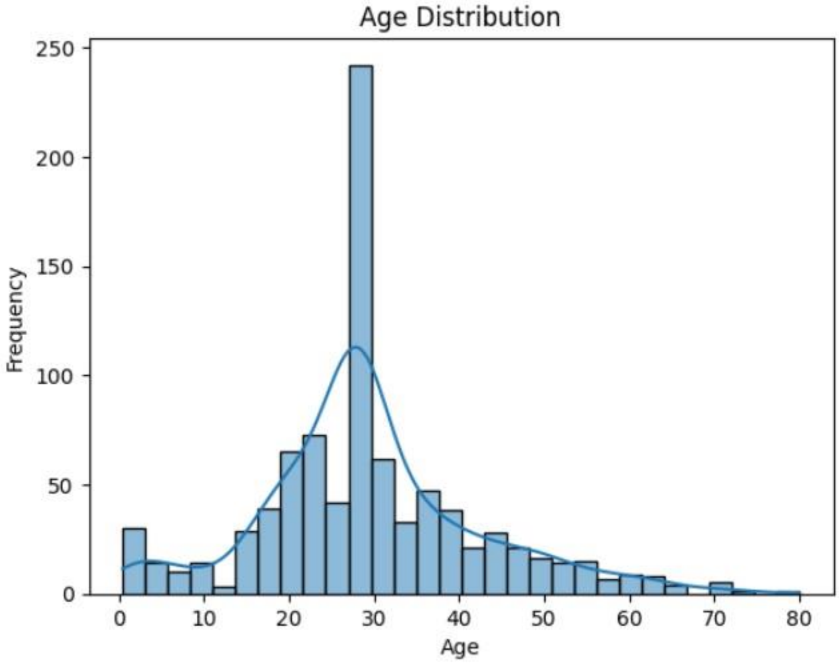
Survival Rate by Class

```python
plt.figure(figsize=(6,5))
sns.countplot(data=data,x="Survived",hue="Pclass")
plt.title("Survival Count by Passenger Class")
plt.show()
```



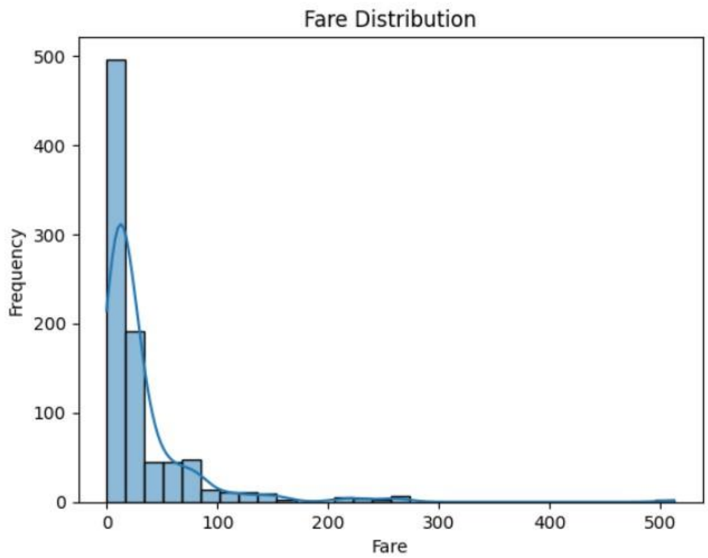
Age Distribution:-

```
#Age Distriution
# Histogram of Age
sns.histplot(data["Age"],bins=30,kde=True)
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.show()
```

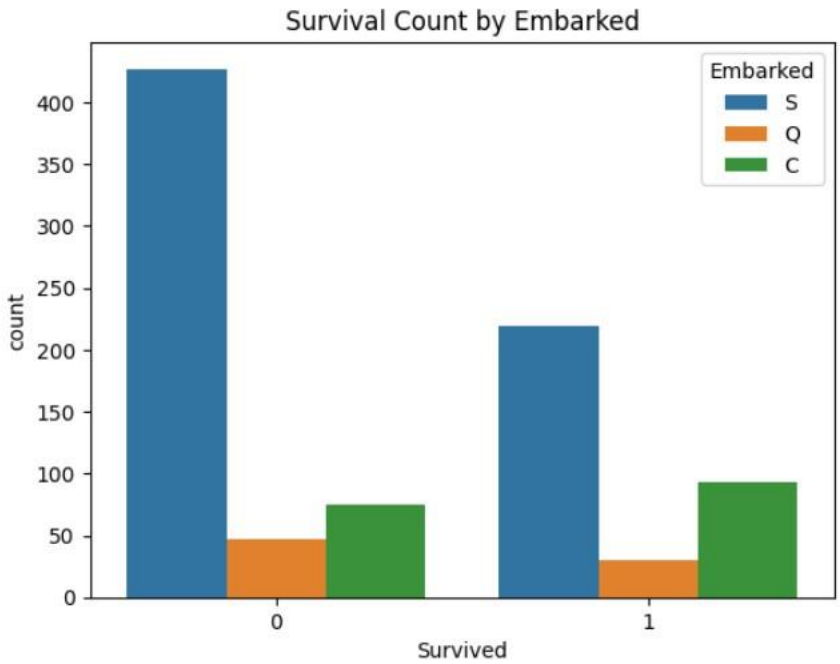


Fare Distribution:-

```
#Fare Distribution
# Histogram of Fare
sns.histplot(data['Fare'], bins=30, kde=True)
plt.title('Fare Distribution')
plt.xlabel('Fare')
plt.ylabel('Frequency')
plt.show()
```

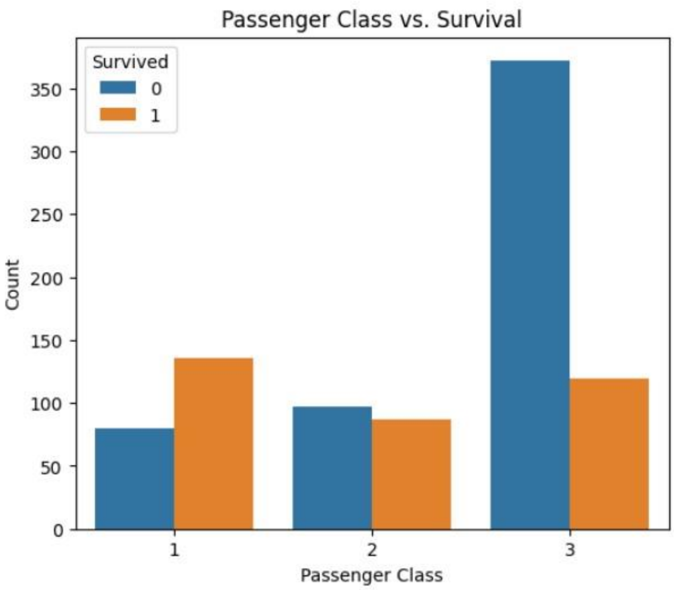


```python
# Bar plot of Survival by Embarked
sns.countplot(x='Survived', hue='Embarked', data=data)
plt.title('Survival Count by Embarked')
plt.show()
```

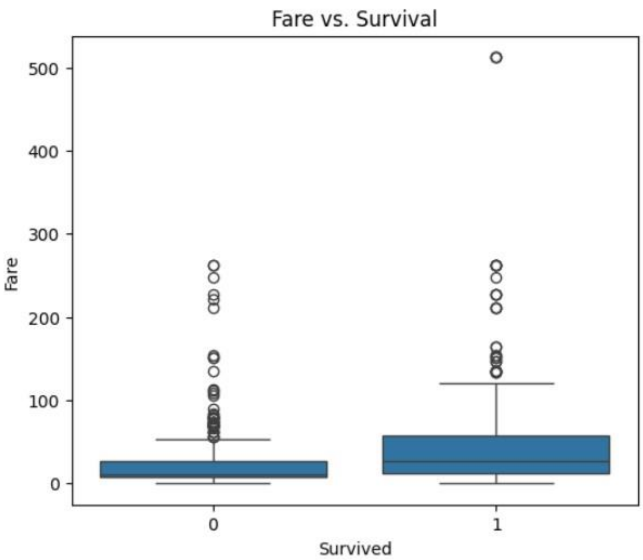



```python
# Violin plot of Age vs. Survival
plt.figure(figsize=(6,5))
sns.violinplot(x='Survived', y='Age', data=data, split=True)
plt.title('Age vs. Survival')
plt.xlabel('Survived')
plt.ylabel('Age')
plt.show()
```

```python
# Bar plot of Passenger Class vs. Survival
plt.figure(figsize=(6,5))
sns.countplot(x='Pclass', hue='Survived', data=data)
plt.title('Passenger Class vs. Survival')
plt.xlabel('Passenger Class')
plt.ylabel('Count')
plt.show()
```




```python
# Box plot of Fare vs. Survival
plt.figure(figsize=(6,5))
sns.boxplot(x='Survived', y='Fare', data=data)
plt.title('Fare vs. Survival')
plt.xlabel('Survived')
plt.ylabel('Fare')
plt.show()
```



This box plot illustrates the relationship between fare and survival status on the Titanic. Here's a detailed explanation:

Chart Components:

1. **X-axis**:

- The x-axis represents the survival status:
    - 0 indicates passengers who did not survive. ▪ 1 indicates passengers who survived.

2. **Y-axis**:

    - The y-axis represents the fare paid by passengers.

3. **Box Plot Elements**:

    - **Box**: The box represents the interquartile range (IQR), which is the range between the first quartile (25th percentile) and the third quartile (75th percentile) of the fare data.
    - **Median Line**: The line inside the box represents the median fare (50th percentile).
    - **Whiskers**: The lines extending from the box represent the range of the data within 1.5 times the IQR from the first and third quartiles.
    - **Outliers**: Points outside the whiskers are considered outliers and are plotted individually.

Insights:

- **Median Fare**:
    - The median fare for passengers who survived (Survived = 1) is higher than for those who did not survive (Survived = 0). This suggests that passengers who paid higher fares had a better chance of survival.
- **Interquartile Range (IQR)**:
    - The IQR for both groups shows the spread of fare values among passengers. Survivors have a wider range of fares compared to non-survivors.
- **Outliers**:
    - There are several outliers in both groups, indicating that some passengers paid significantly higher fares than the majority.

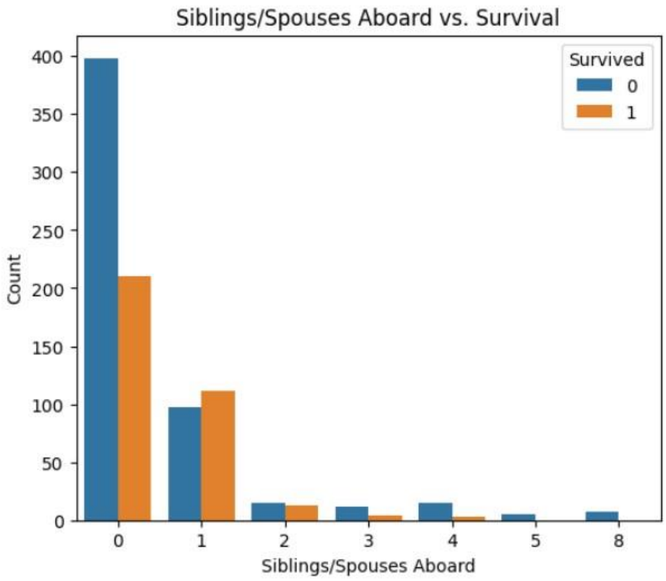
Interpretation:

This box plot reveals a possible correlation between higher fares and higher survival rates on the Titanic. It suggests that wealthier passengers, who could afford higher fares, had a better chance of surviving, possibly due to better access to lifeboats or more favorable locations on the ship.

SibSp (Siblings/Spouses Aboard) vs. Survival

- Investigate how having siblings or spouses aboard affected the survival rate.

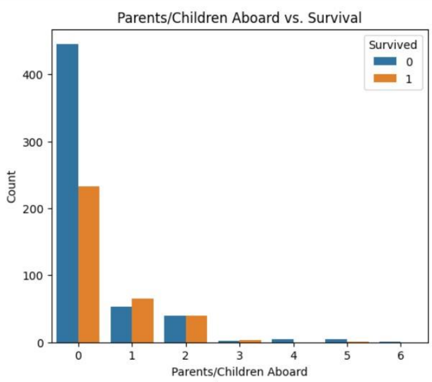
```python
# Bar plot of SibSp vs. Survival
plt.figure(figsize=(6,5))
sns.countplot(data= data,x='SibSp', hue='Survived',)
plt.title('Siblings/Spouses Aboard vs. Survival')
plt.xlabel('Siblings/Spouses Aboard')
plt.ylabel('Count')
plt.show()
```



Parch (Parents/Children Aboard) vs. Survival

- Explore the relationship between having parents or children aboard and the survival rate.

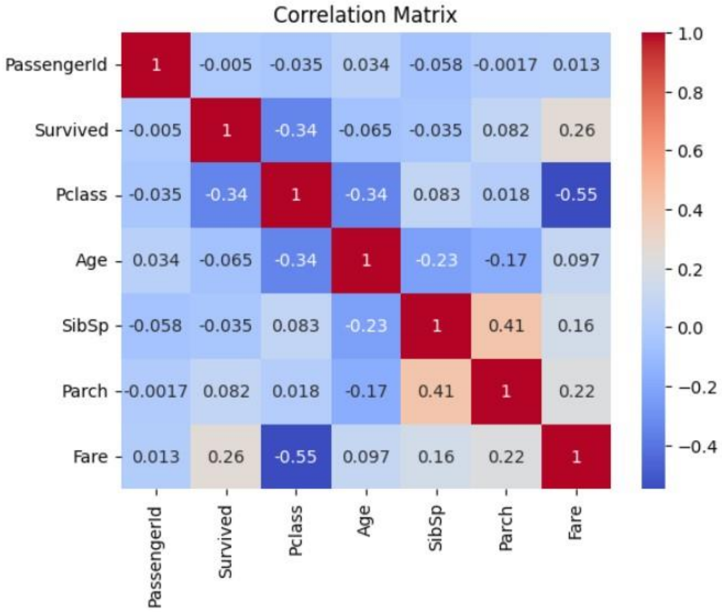
```python
# Bar plot of Parch vs. Survival
plt.figure(figsize=(6,5))
sns.countplot(data=data,x='Parch', hue='Survived')
plt.title('Parents/Children Aboard vs. Survival')
plt.xlabel('Parents/Children Aboard')
plt.ylabel('Count')
plt.show()
```



Correlation Matrix

- Heatmap of Correlation Between Features

```python
# Correlation matrix
numeric_data = data.select_dtypes(include=['number'])
corr_matrix = numeric_data.corr()
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



Conclusion

This project involves cleaning the dataset by handling missing values, performing exploratory data analysis to understand the relationships between variables, and visualizing the patterns and trends in the data. By following these steps, you can gain valuable insights into the Titanic dataset.

Key findings from this analysis include:

- **Passenger Class and Survival**: First-class passengers had a significantly higher survival rate compared to those in second and third class, highlighting the disparity in access to lifeboats and safety.
- **Gender and Survival**: Women had a substantially higher survival rate than men, reflecting the "women and children first" policy during the evacuation.
- **Age and Survival**: Younger passengers, particularly children, showed higher survival rates, emphasizing prioritization during the rescue efforts.
- **Fare and Survival**: Higher fares, indicative of wealth and higher class, correlated with better survival chances.
- **Embarkation Point and Survival**: Passengers who embarked from different ports had varying survival rates, potentially linked to the socioeconomic status associated with each port.