```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/"
directory
# For example, running this (by clicking run or pressing Shift+Enter)
will list all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/)
that gets preserved as output when you create a version using "Save &
Run All"
# You can also write temporary files to /kaggle/temp/, but they won't
be saved outside of the current session
```

```
/kaggle/input/titanic/train.csv
/kaggle/input/titanic/test.csv
/kaggle/input/titanic/gender_submission.csv
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results__.
html
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__notebook__
.ipynb
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__output__.j
son
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/custom.css
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___80_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___72_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___60_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___70_1.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___45_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___56_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___58_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___99_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___43_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___
files/__results___30_1.png
```

```
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___64_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___62_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___66_1.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___41_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___96_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___50_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___48_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___94_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___92_0.png
/kaggle/input/masterclass-1-a-comprehensive-guide-for-eda/__results___files/__results___74_0.png
```

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

# Load dataset
df = pd.read_csv('/kaggle/input/titanic/train.csv')

# Basic exploration
print(df.head())
print(df.info())
print(df.describe())
print(f"Dataset shape: {df.shape}")
```

```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age
SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0
1
1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0
1
2                             Heikkinen, Miss. Laina  female  26.0
0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0
```

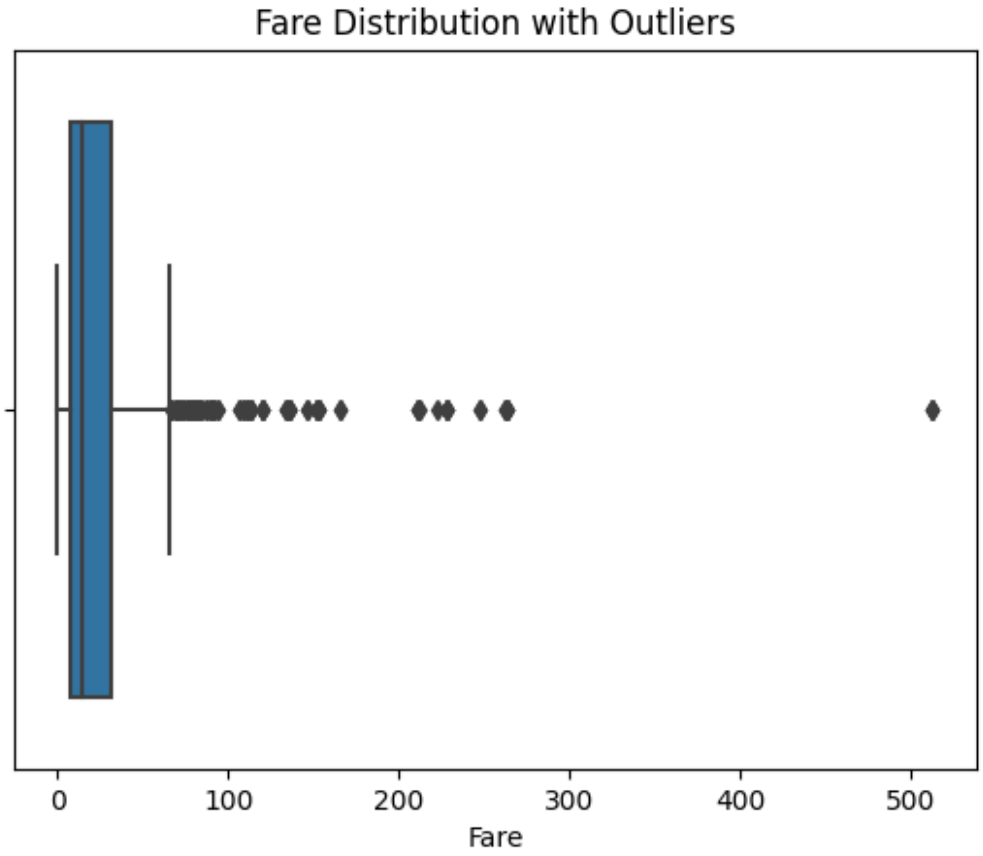
```
1
4                            Allen, Mr. William Henry    male  35.0
0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S
<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
None
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
```

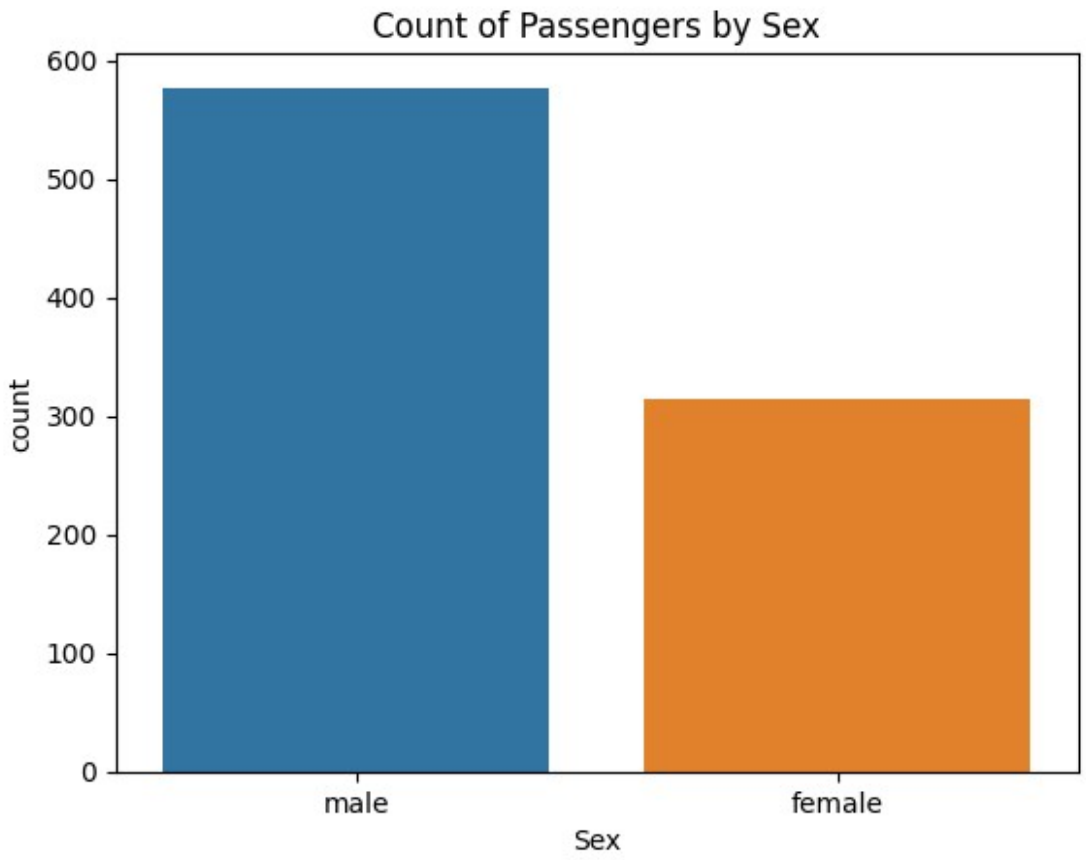
```
max      6.000000  512.329200
Dataset shape: (891, 12)
```

```
# Boxplot to detect outliers
sns.boxplot(x=df['Fare'])
plt.title("Fare Distribution with Outliers")
plt.show()
```

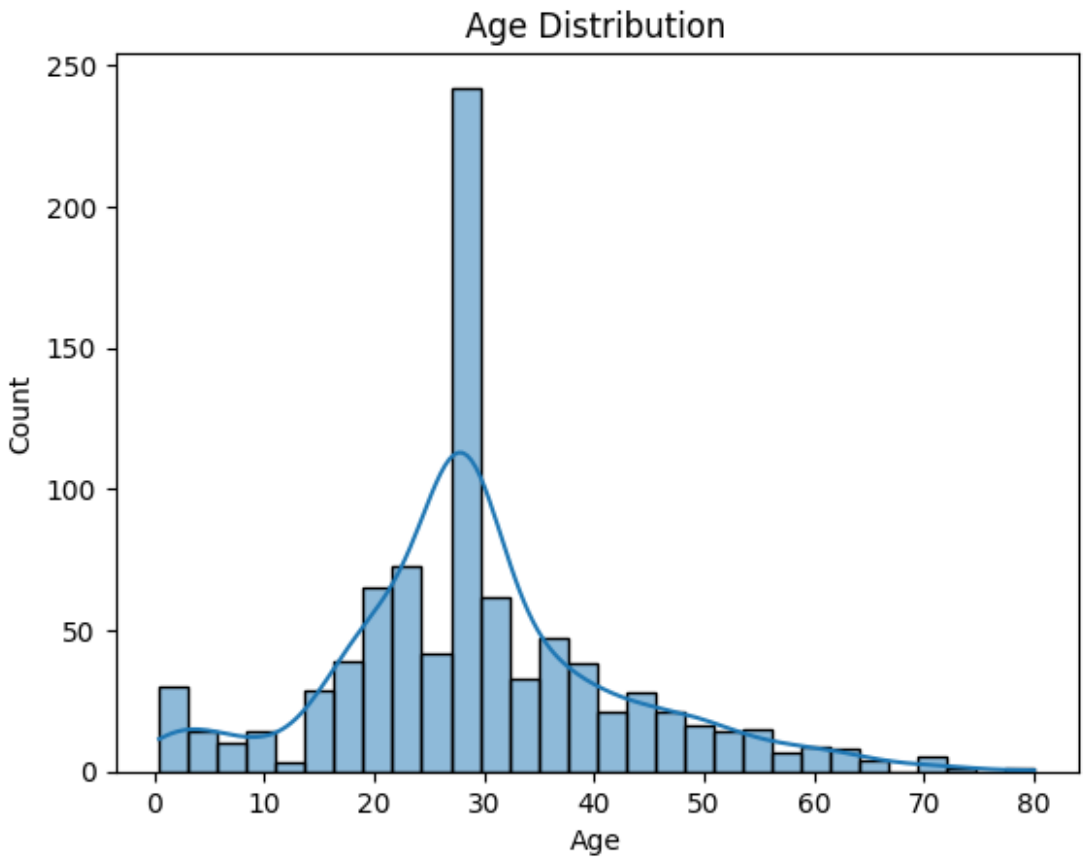



```
# Categorical feature: Sex
sns.countplot(x='Sex', data=df)
plt.title("Count of Passengers by Sex")
plt.show()

# Continuous feature: Age
sns.histplot(df['Age'], bins=30, kde=True)
plt.title("Age Distribution")
plt.show()
```



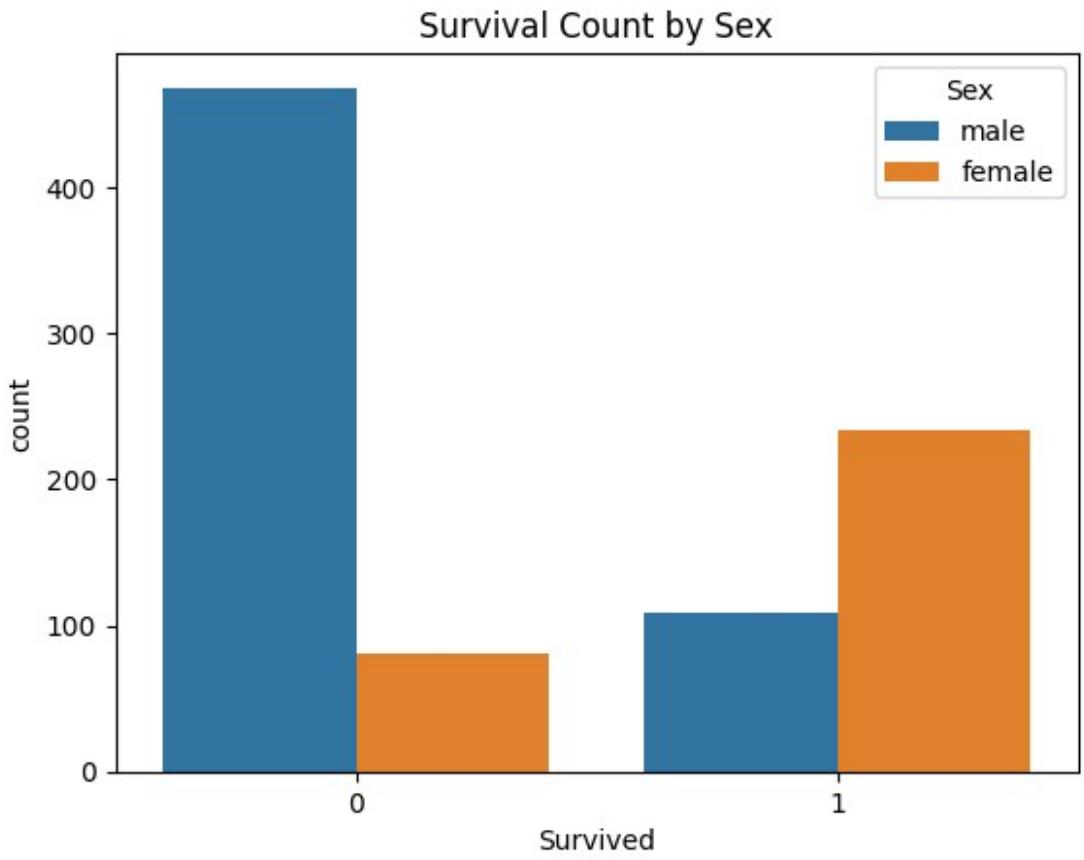

```
/usr/local/lib/python3.11/dist-packages/seaborn/_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



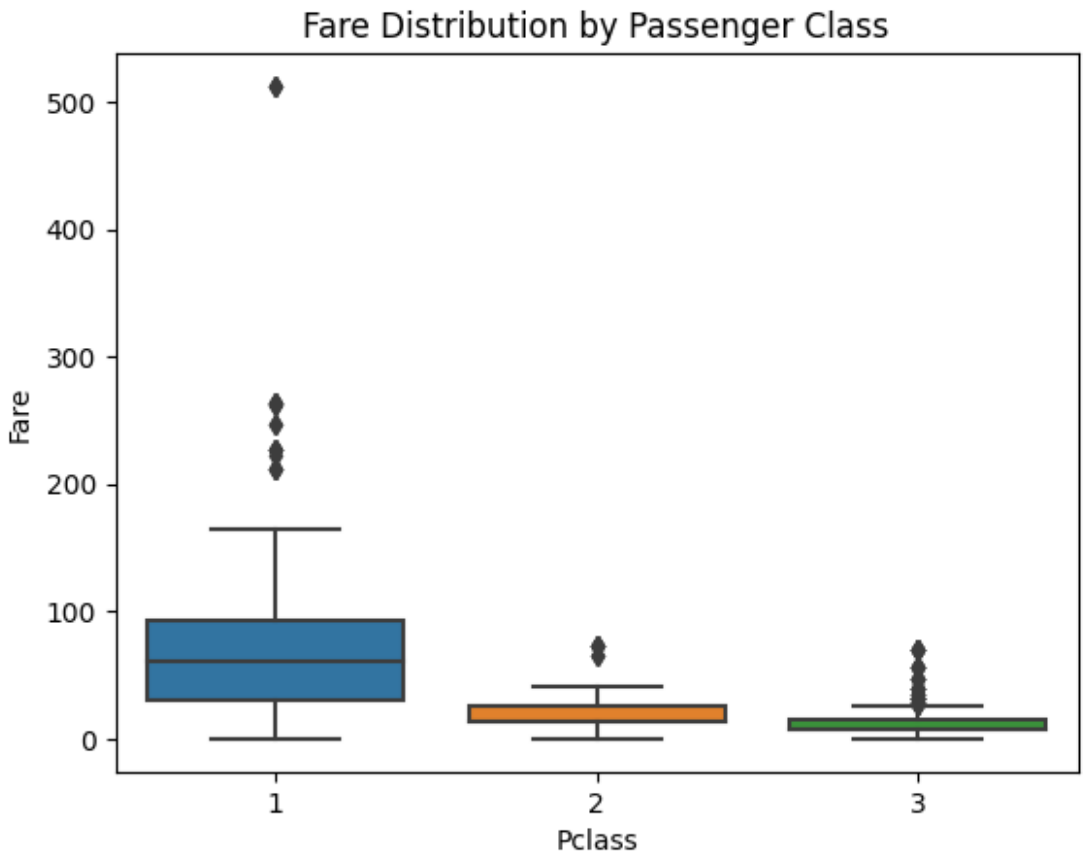

```
# Survival by Sex
sns.countplot(x='Survived', hue='Sex', data=df)
plt.title("Survival Count by Sex")
plt.show()

# Age distribution by survival
sns.violinplot(x='Survived', y='Age', data=df)
plt.title("Age vs Survival")
plt.show()

# Fare vs Pclass
sns.boxplot(x='Pclass', y='Fare', data=df)
plt.title("Fare Distribution by Passenger Class")
plt.show()
```

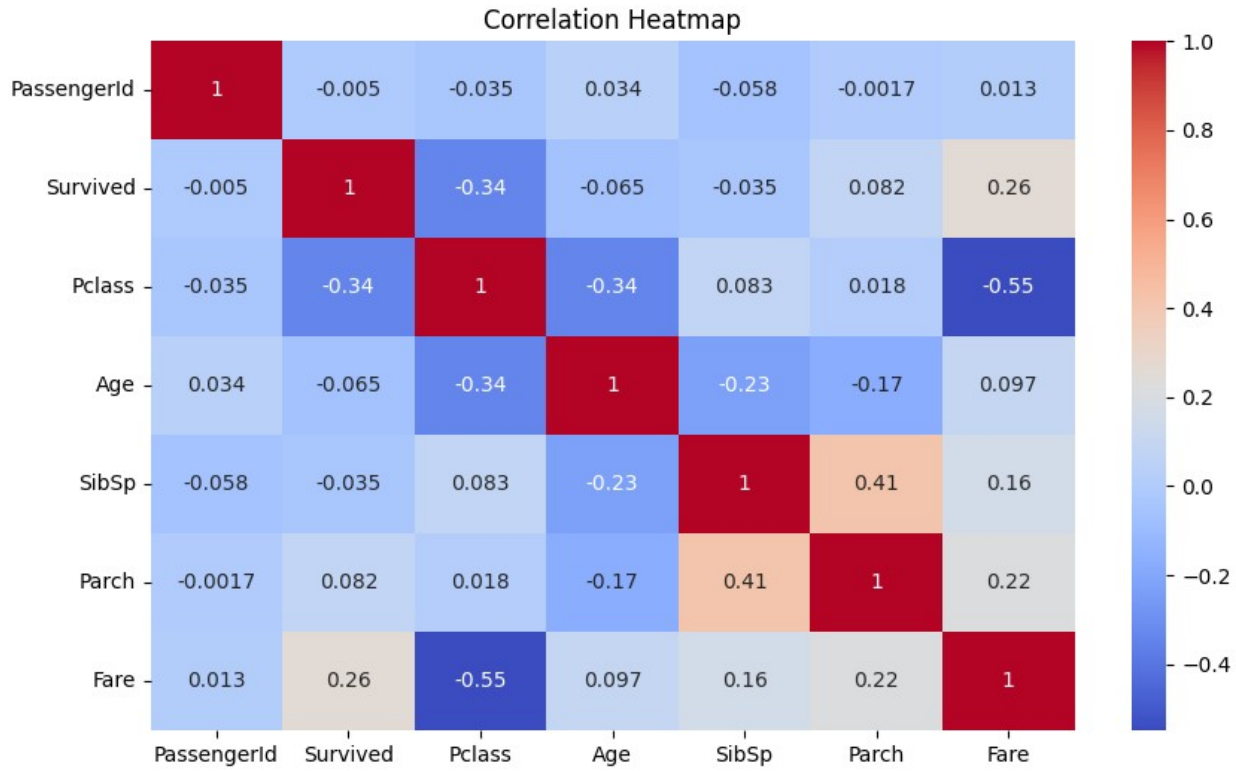

Survival Count by Sex
Sex
male
female
400
300
200
100
0
count
0
1
Survived


Age vs Survival
80
60
40
20
0
Age
0
1
Survived




```
# Select only numeric columns
numeric_df = df.select_dtypes(include='number')

# Plot correlation heatmap
plt.figure(figsize=(10,6))
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```

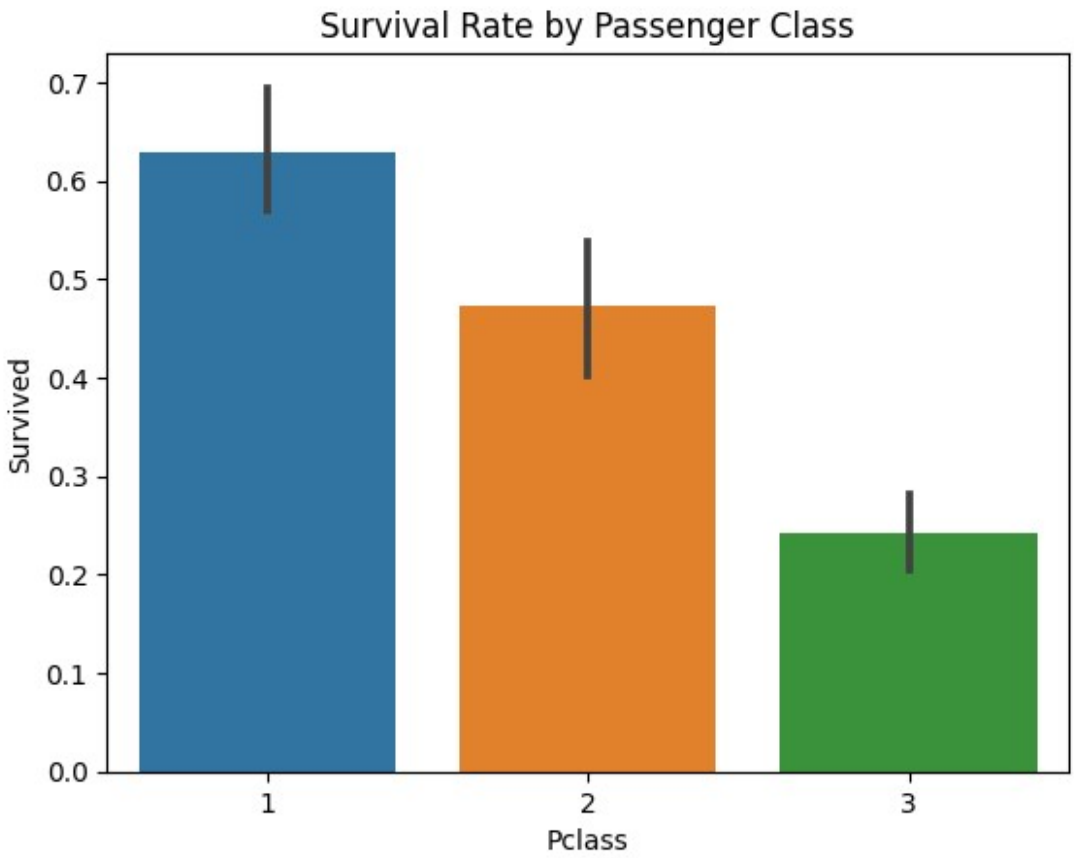


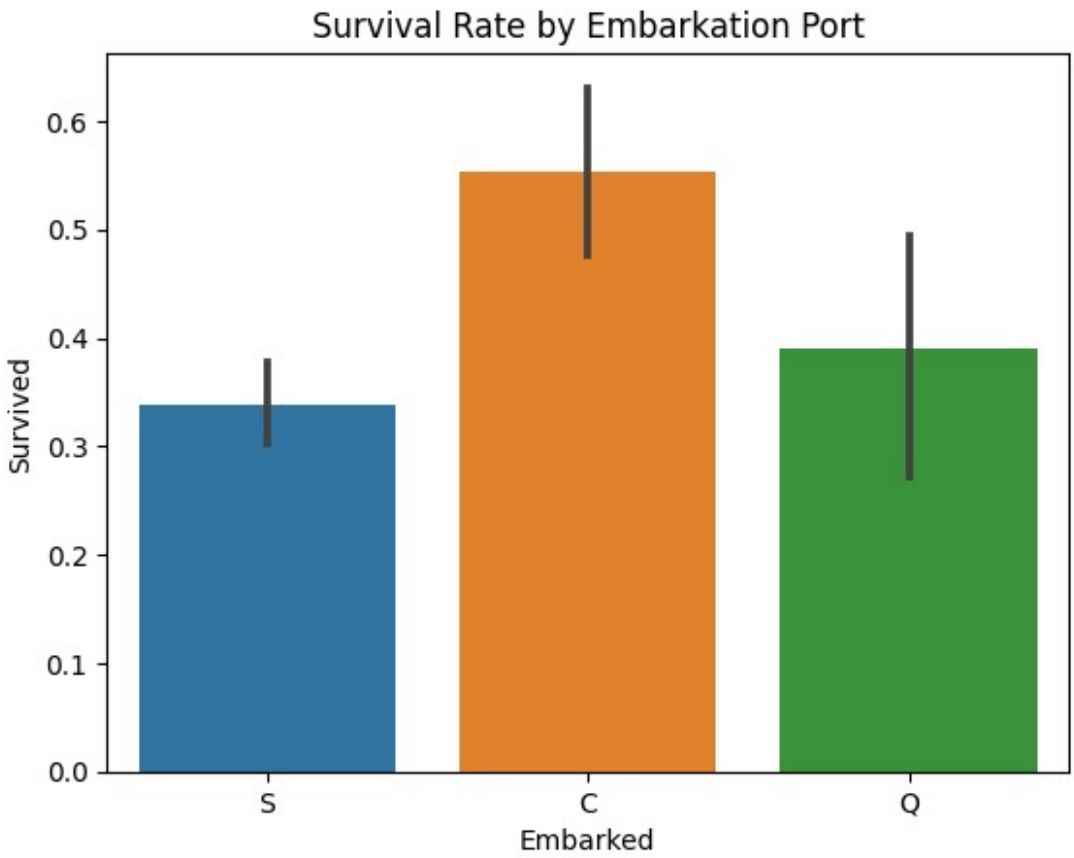

```
# Survival rate by class
print(df.groupby('Pclass')['Survived'].mean())

# Visualize
sns.barplot(x='Pclass', y='Survived', data=df)
plt.title("Survival Rate by Passenger Class")
plt.show()

# Survival rate by Embarked
sns.barplot(x='Embarked', y='Survived', data=df)
plt.title("Survival Rate by Embarkation Port")
plt.show()
```

```
Pclass
1    0.629630
2    0.472826
3    0.242363
Name: Survived, dtype: float64
```


Survival Rate by Passenger Class
Survived
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
1
2
3
Pclass


Survival Rate by Embarkation Port
Survived
0.0
0.1
0.2
0.3
0.4
0.5
0.6
S
C
Q
Embarked