Task2

November 17, 2025

1 Task 2

[384]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

plt.style.use('default')
sns.set_theme()

df = pd.read_csv("Titanic_Dataset.csv")
df.head()
```

[384]:

```
   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  
```

[385]:

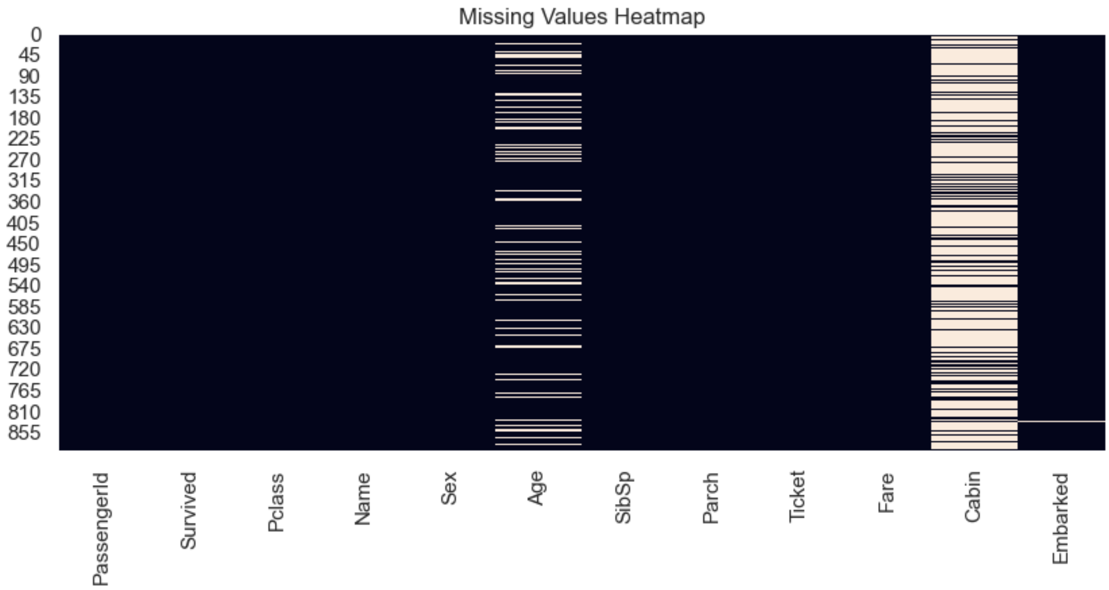
```
df.info()
df.shape
df.describe()
print(df.value_counts())
```

```
df.isnull().sum()
```

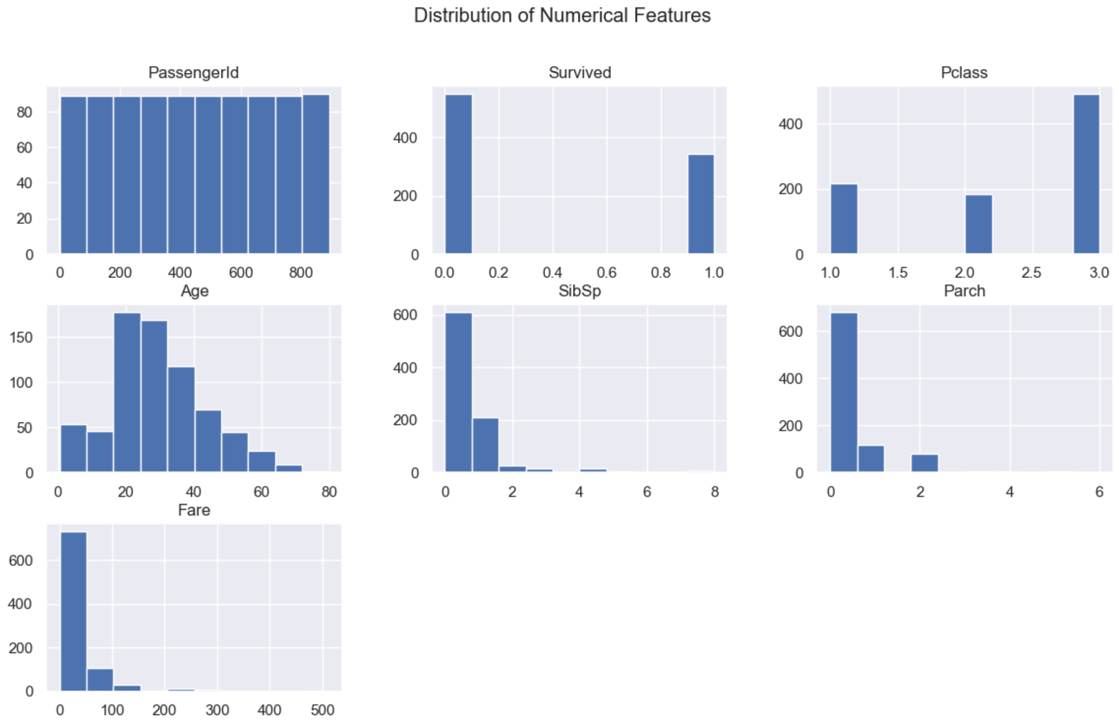
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    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
PassengerId  Survived  Pclass  Name
Sex     Age   SibSp  Parch  Ticket    Fare     Cabin        Embarked
2            1         1       Cumings, Mrs. John Bradley (Florence Briggs
Thayer)  female  38.0  1      0      PC 17599  71.2833  C85          C
1
4            1         1       Futrelle, Mrs. Jacques Heath (Lily May Peel)
female  35.0  1      0      113803    53.1000  C123         S           1
7            0         1       McCarthy, Mr. Timothy J
male    54.0  0      0      17463     51.8625  E46          S           1
11           1         3       Sandstrom, Miss. Marguerite Rut
female  4.0   1      1      PP 9549   16.7000  G6           S           1
12           1         1       Bonnell, Miss. Elizabeth
female  58.0  0      0      113783    26.5500  C103         S           1
                                                                           ..
872          1         1       Beckwith, Mrs. Richard Leonard (Sallie Monypeny)
female  47.0  1      1      11751     52.5542  D35          S           1
873          0         1       Carlsson, Mr. Frans Olof
male    33.0  0      0      695       5.0000   B51 B53 B55  S           1
880          1         1       Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)
female  56.0  0      1      11767     83.1583  C50          C           1
888          1         1       Graham, Miss. Margaret Edith
female  19.0  0      0      112053    30.0000  B42          S           1
890          1         1       Behr, Mr. Karl Howell
male    26.0  0      0      111369    30.0000  C148         C           1
Name: count, Length: 183, dtype: int64
```

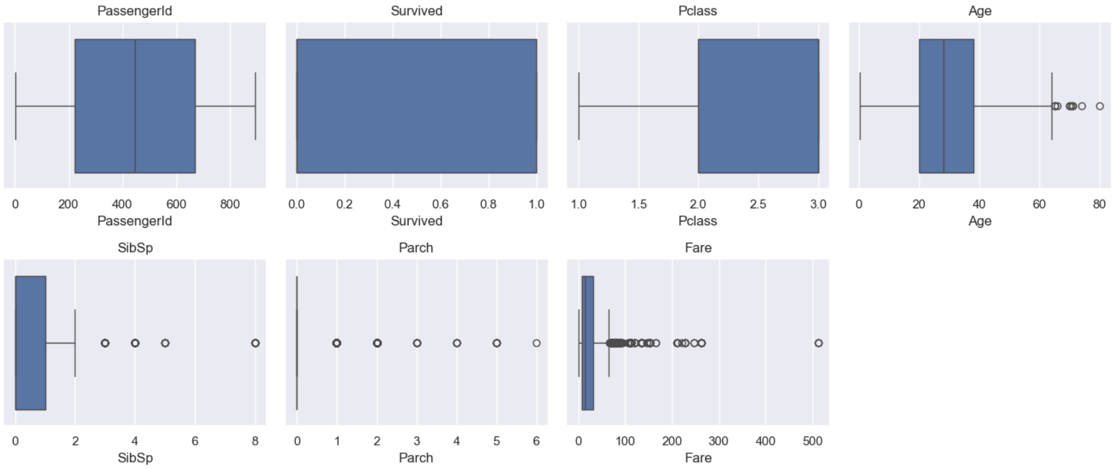
[385]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

[386]:
```
plt.figure(figsize=(10,4))
sns.heatmap(df.isnull(), cbar=False)
plt.title("Missing Values Heatmap")
plt.show()
```

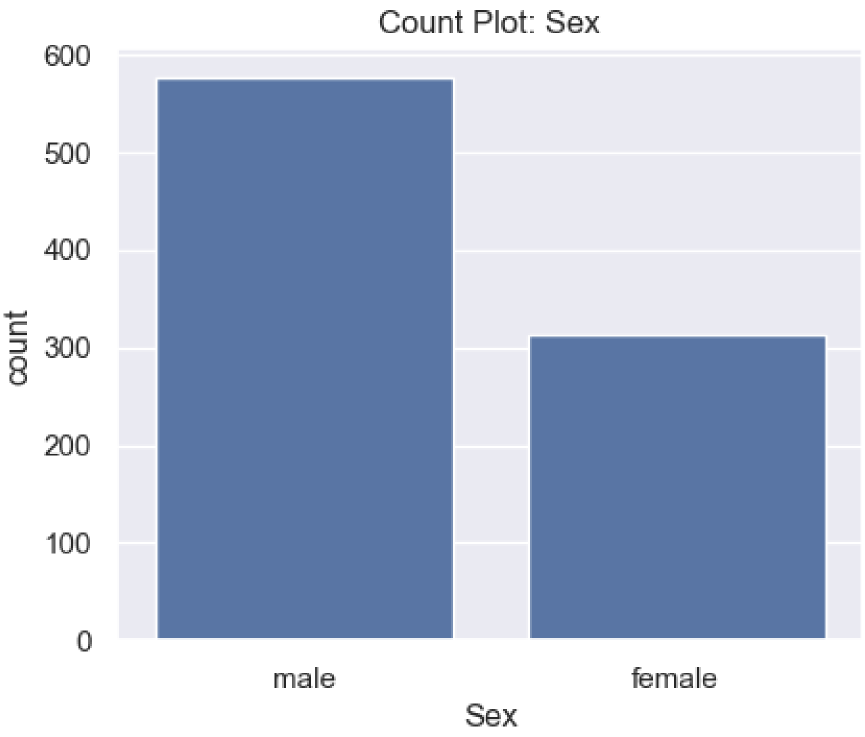


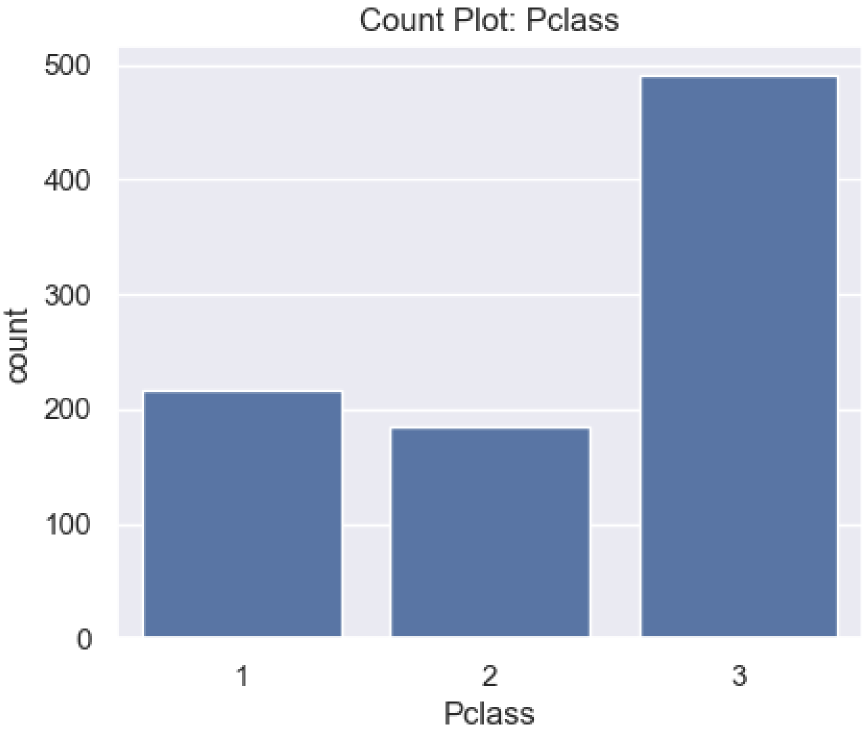
[387]:
```
numeric_cols = df.select_dtypes(include=['int64','float64']).columns

df[numeric_cols].hist(figsize=(14,8))
plt.suptitle("Distribution of Numerical Features")
plt.show()

plt.figure(figsize=(14,6))
for i, col in enumerate(numeric_cols, 1):
    plt.subplot(2, 4, i)
    sns.boxplot(x=df[col])
    plt.title(col)
plt.tight_layout()
plt.show()
```

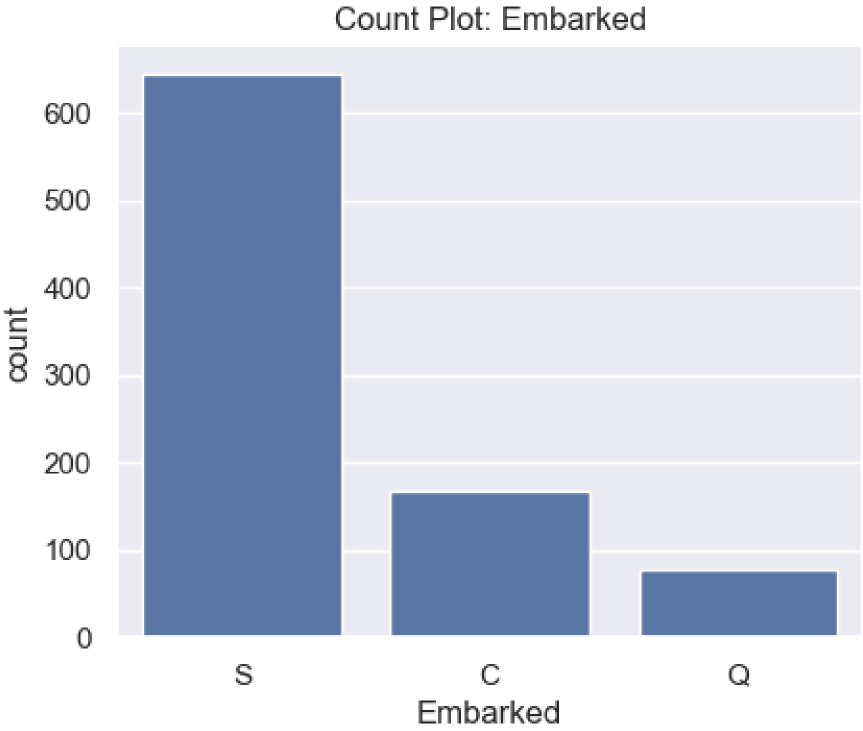


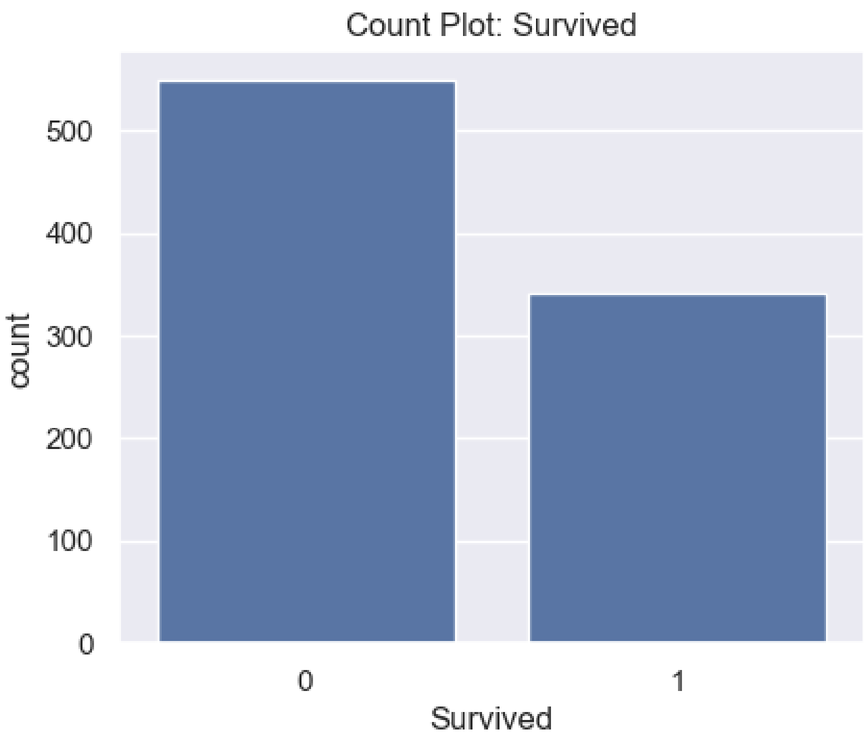


```
[388]: categorical_cols = ['Sex', 'Pclass', 'Embarked', 'Survived']
       for col in categorical_cols:
           plt.figure(figsize=(5,4))
           sns.countplot(x=df[col])
           plt.title(f"Count Plot: {col}")
           plt.show()
```



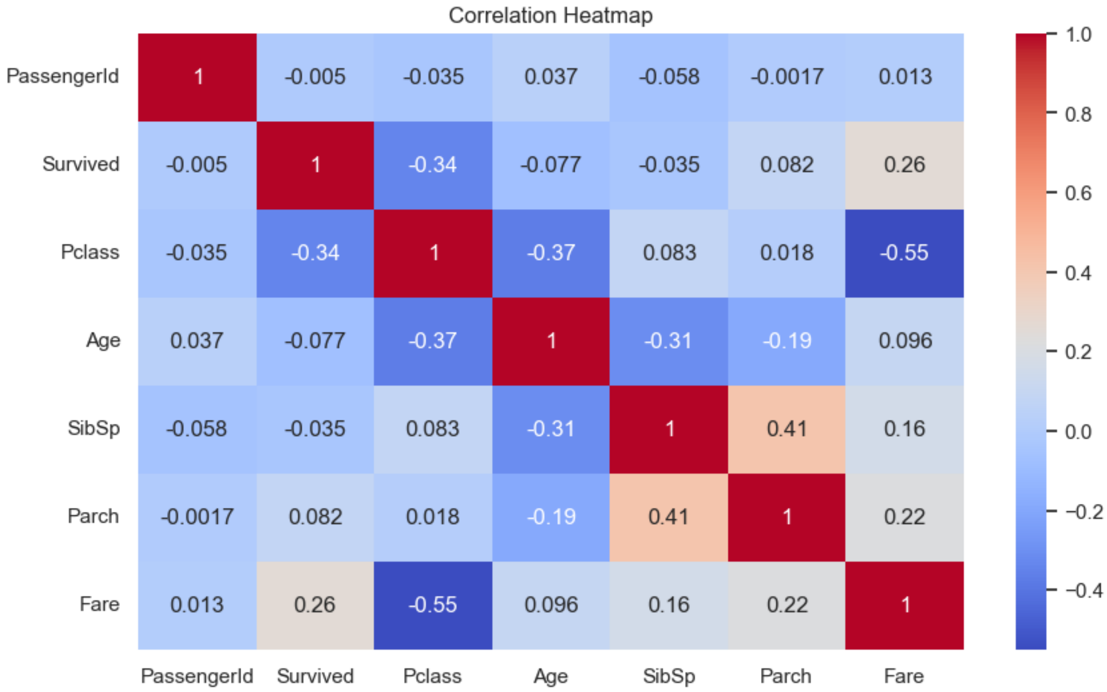

Count Plot: Pclass
500
400
300
200
100
0
count
1
2
3
Pclass


Count Plot: Embarked
600
500
400
300
200
100
0
count
S
C
Q
Embarked



[389]:

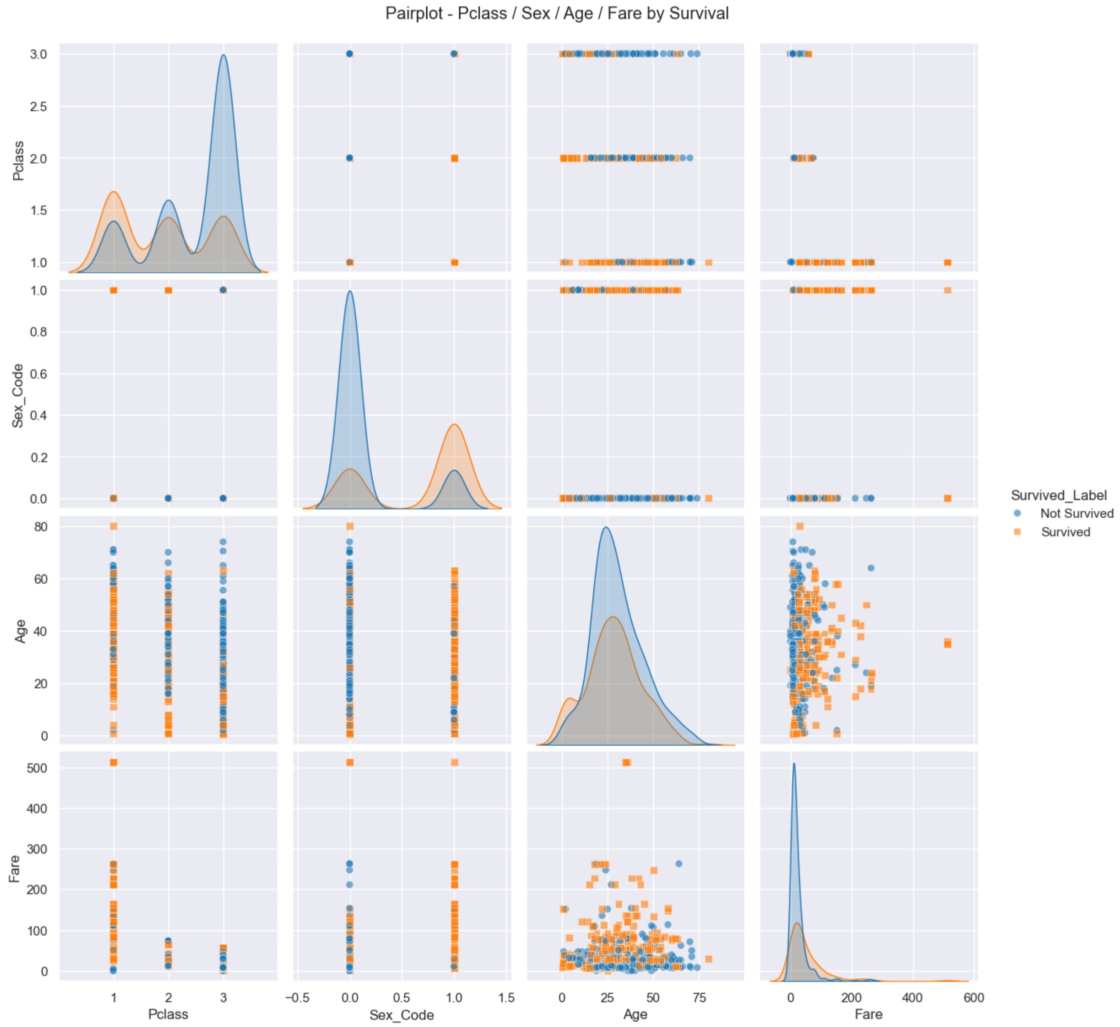
```
plt.figure(figsize=(10,6))
sns.heatmap(df.select_dtypes(include=['int64','float64']).corr(), annot=True,␣
  ↪cmap="coolwarm")
plt.title("Correlation Heatmap")
plt.show()
```



	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1	-0.005	-0.035	0.037	-0.058	-0.0017	0.013
Survived	-0.005	1	-0.34	-0.077	-0.035	0.082	0.26
Pclass	-0.035	-0.34	1	-0.37	0.083	0.018	-0.55
Age	0.037	-0.077	-0.37	1	-0.31	-0.19	0.096
SibSp	-0.058	-0.035	0.083	-0.31	1	0.41	0.16
Parch	-0.0017	0.082	0.018	-0.19	0.41	1	0.22
Fare	0.013	0.26	-0.55	0.096	0.16	0.22	1

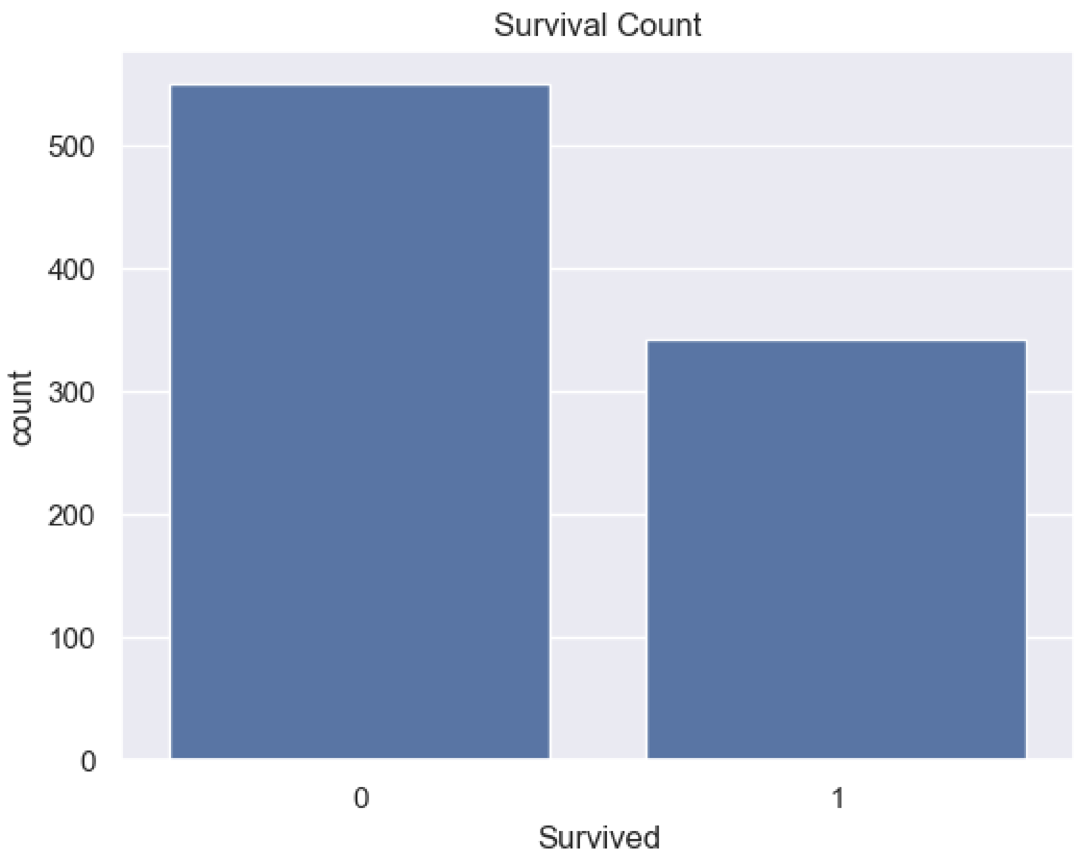
[390]:

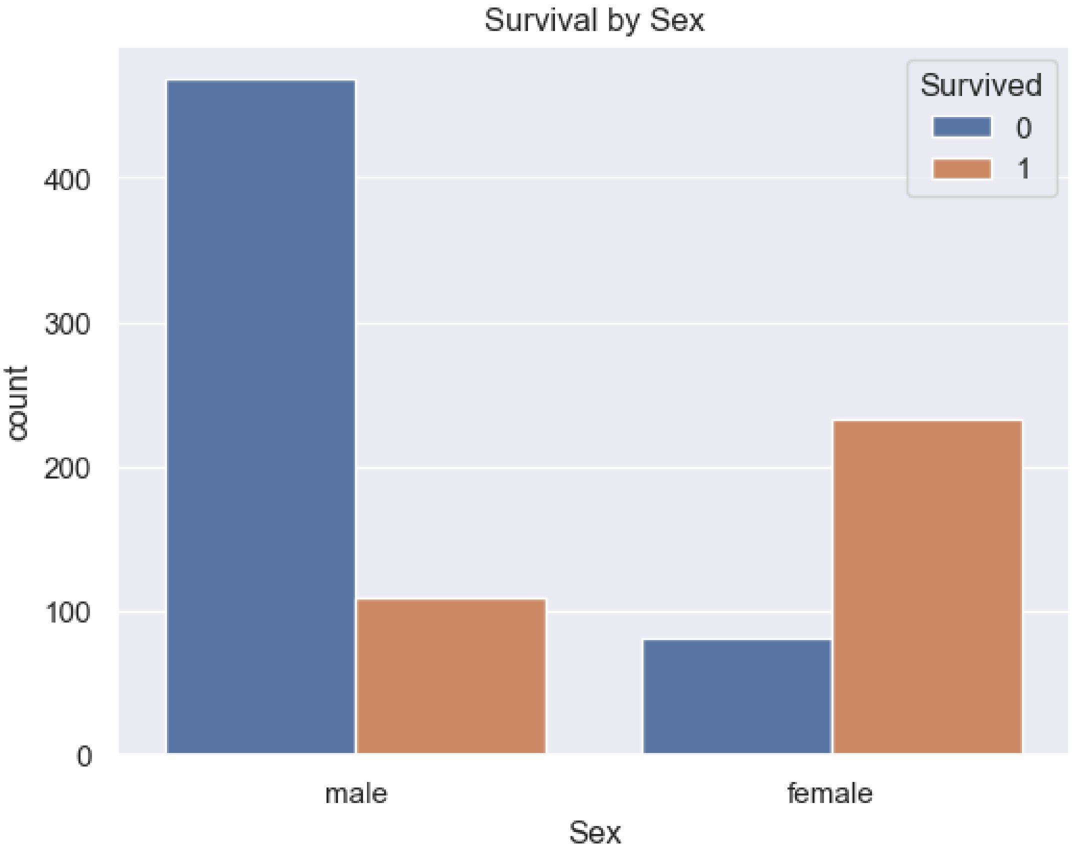
```
df['Survived_Label'] = df['Survived'].map({0: 'Not Survived', 1: 'Survived'})
df['Sex_Code'] = df['Sex'].map({'male': 0, 'female': 1})

vars_to_plot = ['Pclass', 'Sex_Code', 'Age', 'Fare']
df_pp = df[['Survived_Label'] + vars_to_plot].dropna()

palette = {'Not Survived': 'tab:blue', 'Survived': 'tab:orange'}

g = sns.pairplot(
    df_pp,hue='Survived_Label',
    vars=vars_to_plot,
    palette=palette,
    diag_kind='kde',
    plot_kws={'alpha': 0.6, 's': 40},
    diag_kws={'fill': True},
    markers=['o', 's'],
    height=3,
    aspect=1
)

g.figure.suptitle("Pairplot - Pclass / Sex / Age / Fare by Survival", y=1.02)
plt.show()
```

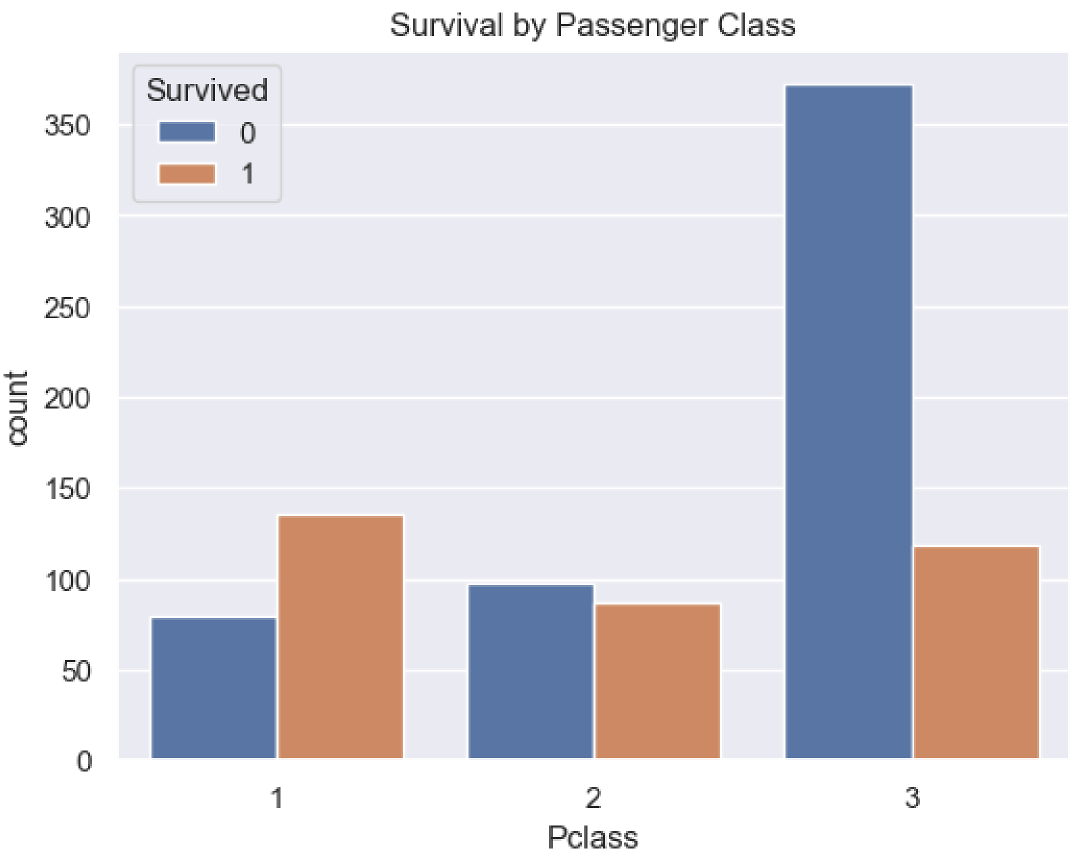


[391]:

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

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

sns.countplot(x='Pclass', hue='Survived', data=df)
plt.title("Survival by Passenger Class")
plt.show()
```


Survival Count
count
500
400
300
200
100
0
0
1
Survived


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



[392]:

```
df.groupby("Pclass")["Survived"].mean()
df.groupby("Sex")["Survived"].mean()
df.groupby(["Pclass","Sex"])["Survived"].mean()
```

[392]:

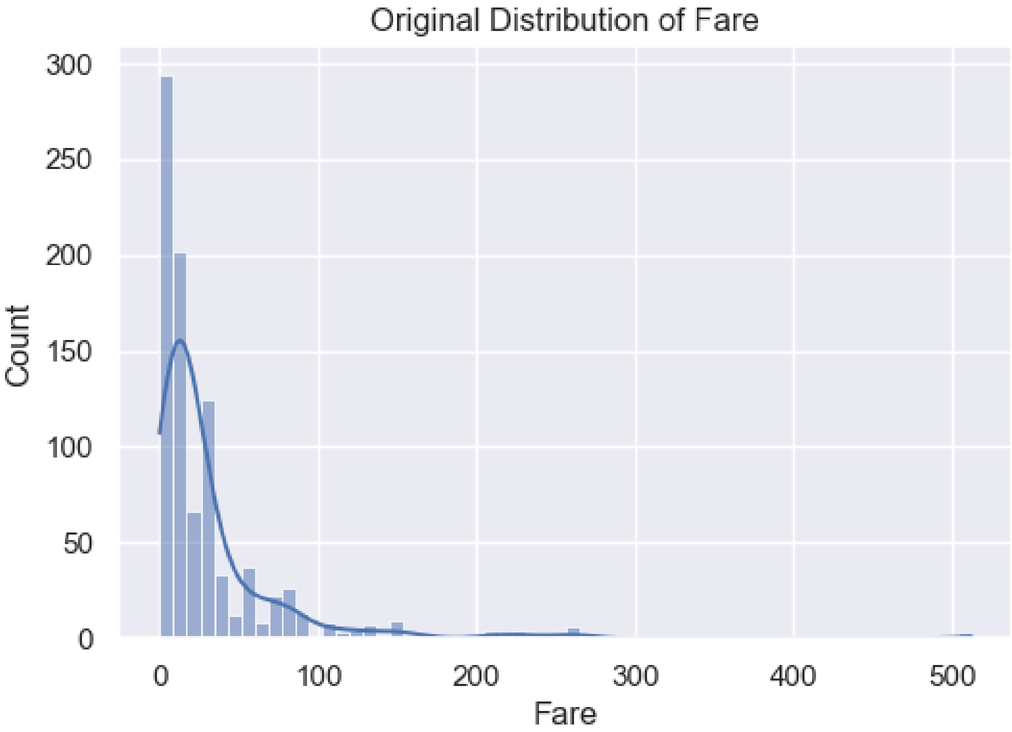
```
Pclass  Sex
1       female    0.968085
        male      0.368852
2       female    0.921053
        male      0.157407
3       female    0.500000
        male      0.135447
Name: Survived, dtype: float64
```

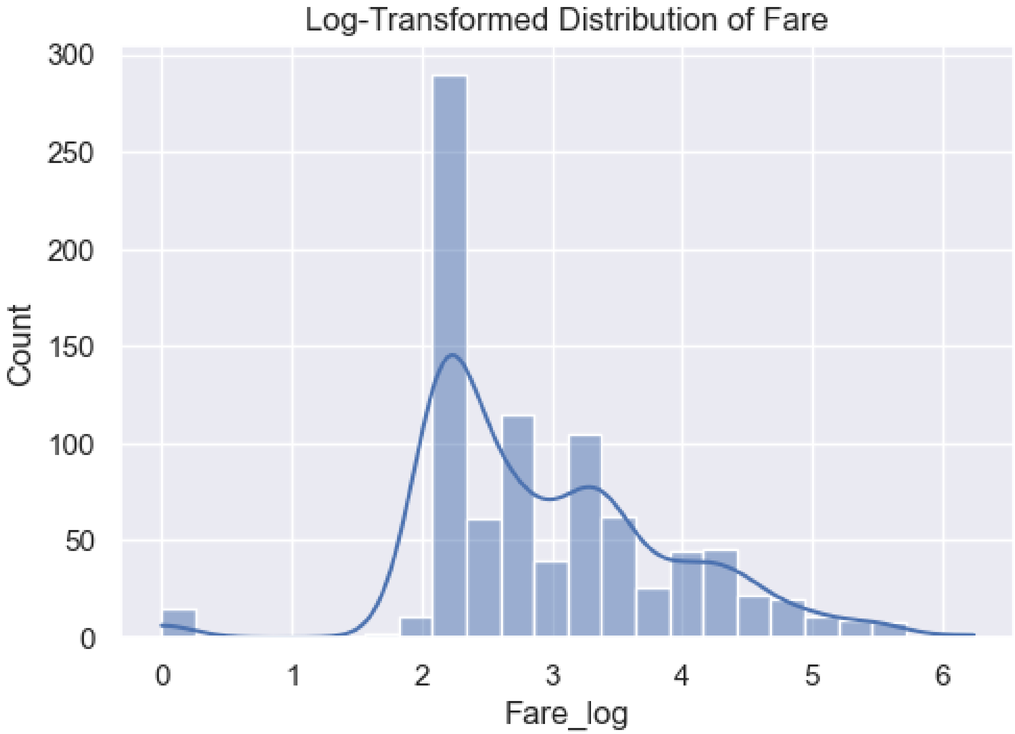
[393]:

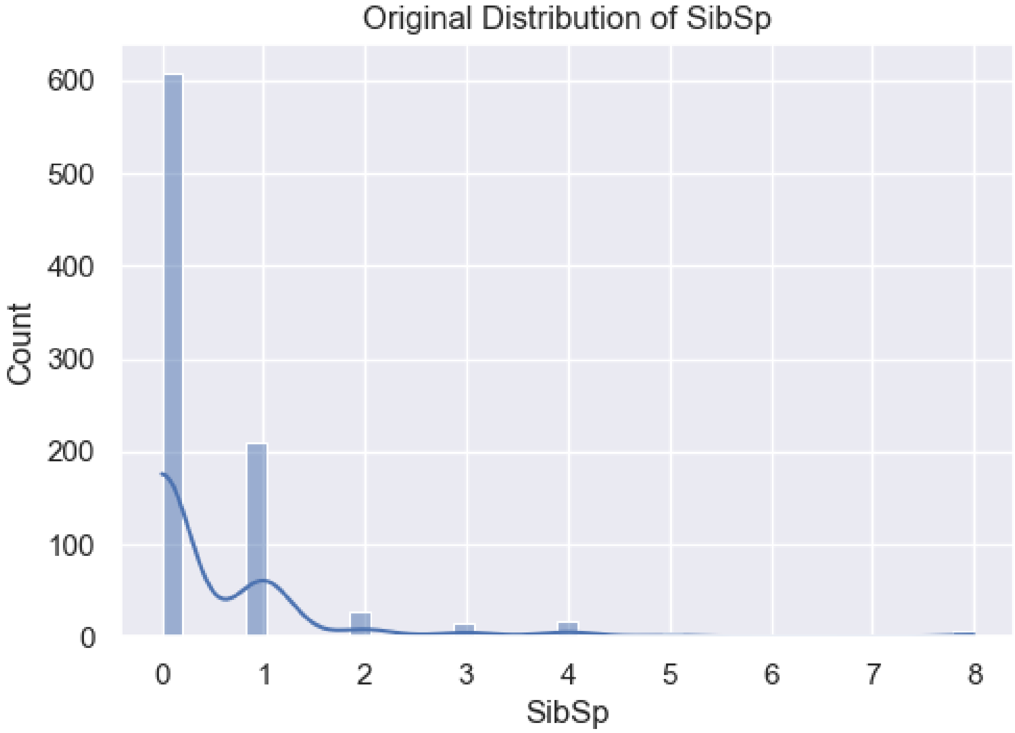
```
print("Skewness:\n", df[numeric_cols].skew().sort_values(ascending=False))

log_transform_cols = ['Fare', 'SibSp', 'Parch', 'Age']

for col in log_transform_cols:
    df[col + '_log'] = np.log1p(df[col])
```

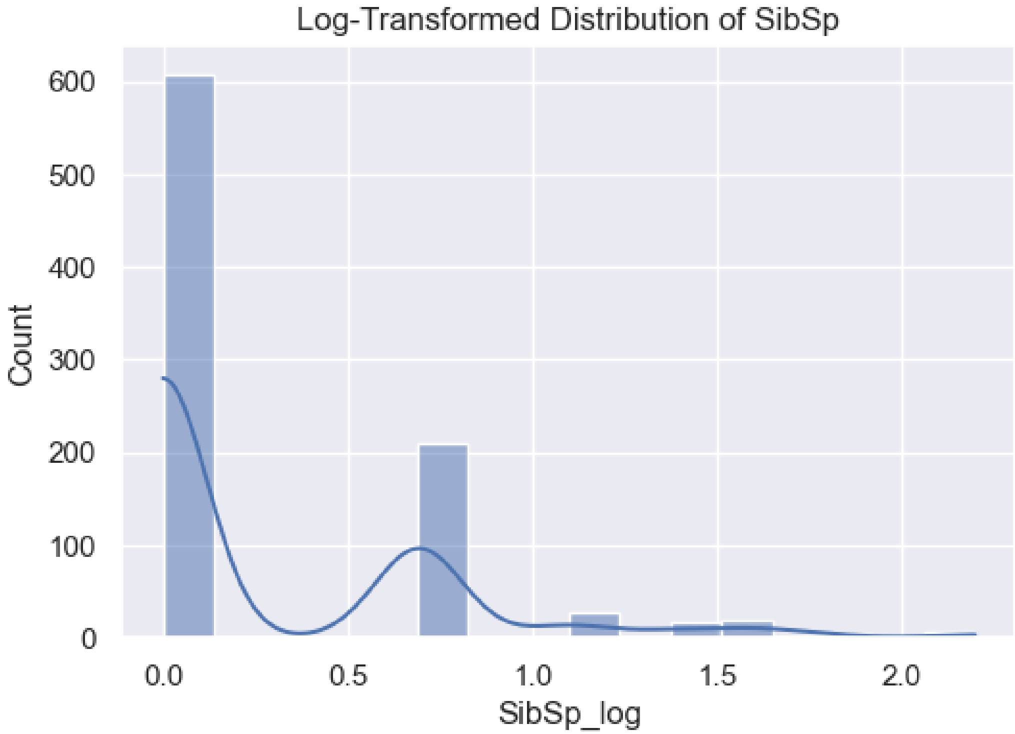
```
for col in log_transform_cols:
    plt.figure(figsize=(6,4))
    sns.histplot(df[col], kde=True)
    plt.title(f"Original Distribution of {col}")
    plt.xlabel(col)
    plt.ylabel("Count")
    plt.show()

    plt.figure(figsize=(6,4))
    sns.histplot(df[col + '_log'], kde=True)
    plt.title(f"Log-Transformed Distribution of {col}")
    plt.xlabel(col + '_log')
    plt.ylabel("Count")
    plt.show()
```

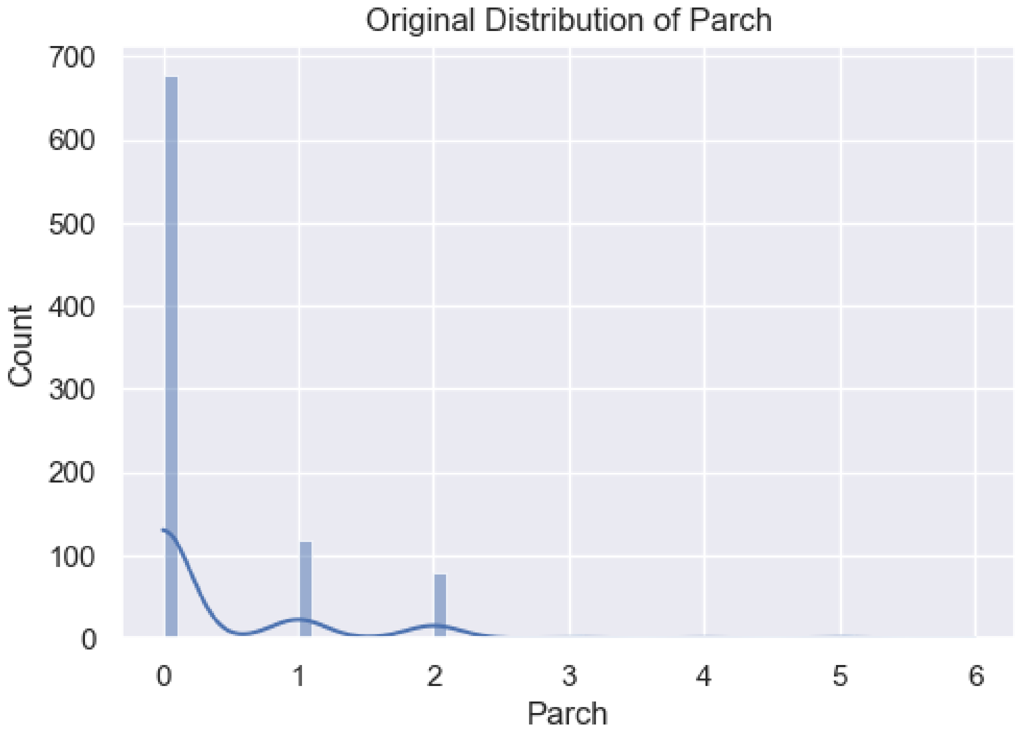
```
Skewness:
 Fare           4.787317
SibSp          3.695352
Parch          2.749117
Survived       0.478523
Age            0.389108
PassengerId    0.000000
Pclass        -0.630548
dtype: float64
```

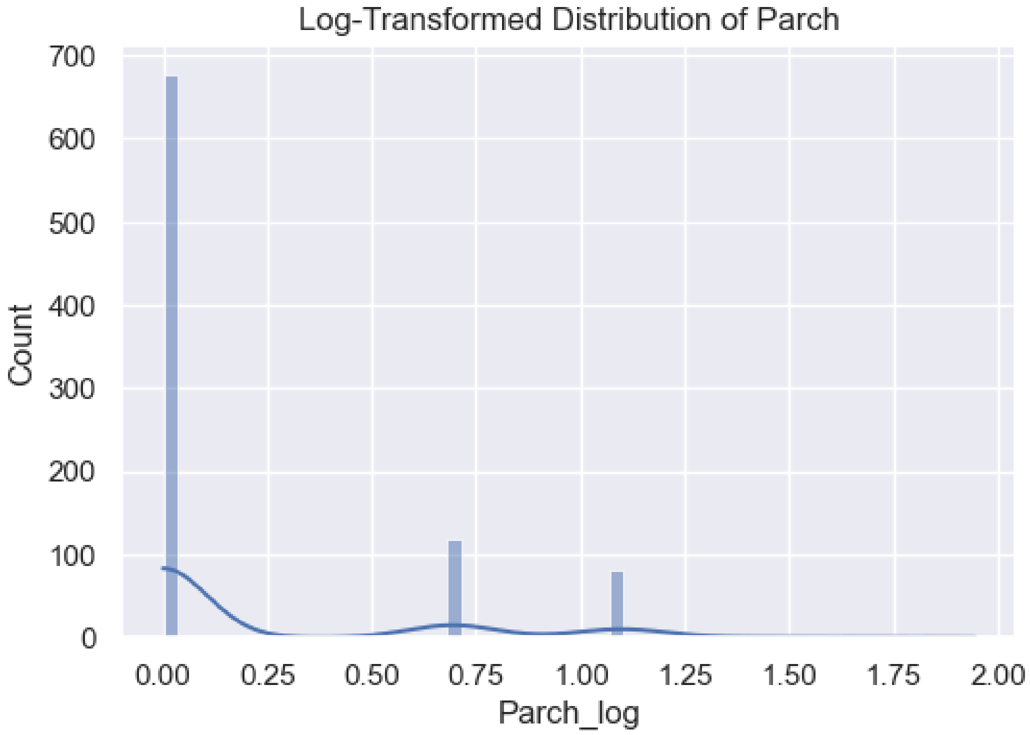


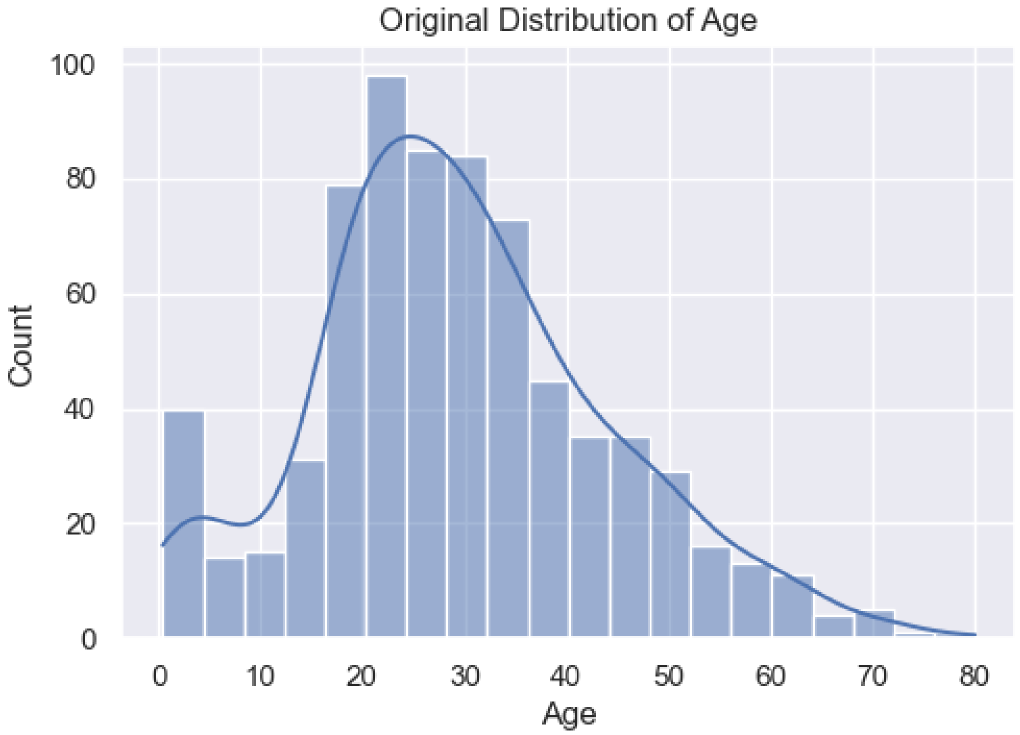

Log-Transformed Distribution of Fare
300
250
200
150
100
50
0
Count
0
1
2
3
4
5
6
Fare_log

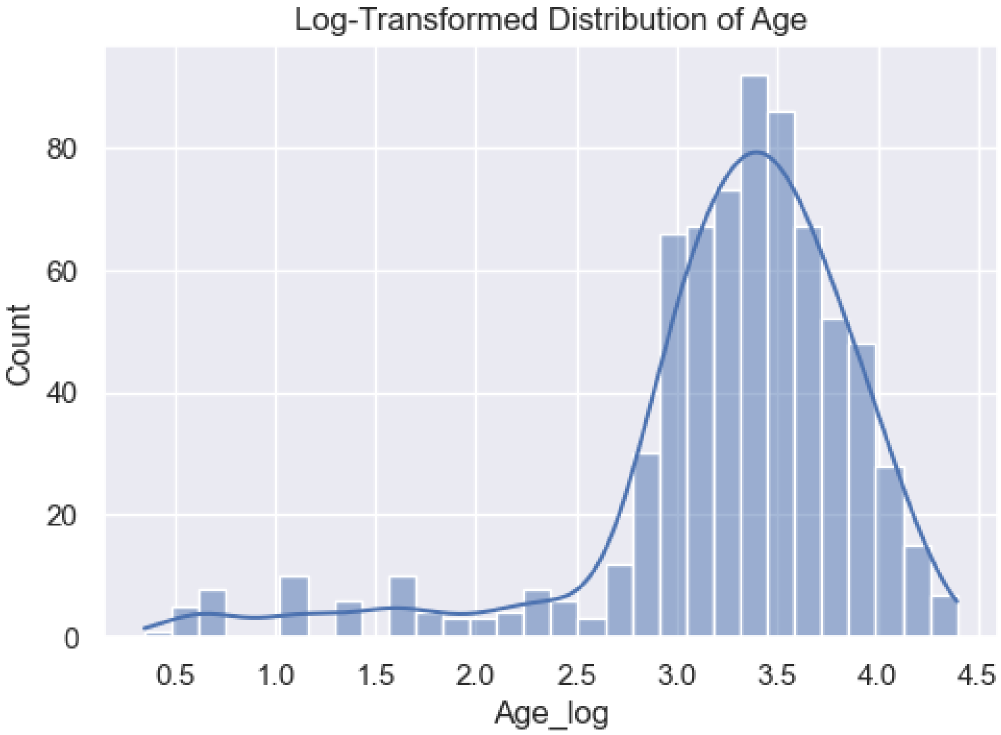

Original Distribution of SibSp
600
500
400
300
200
100
0
Count
0
1
2
3
4
5
6
7
8
SibSp


Log-Transformed Distribution of SibSp
Count
0
100
200
300
400
500
600
0.0
0.5
1.0
1.5
2.0
SibSp_log

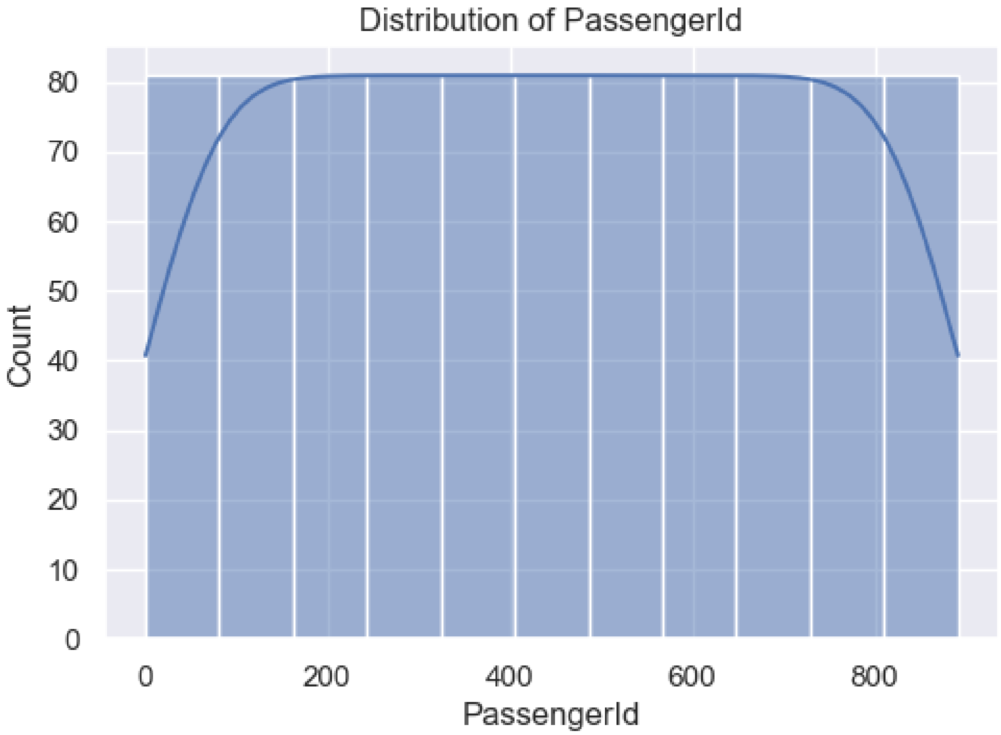

Original Distribution of Parch
Count
0
100
200
300
400
500
600
700
0
1
2
3
4
5
6
Parch

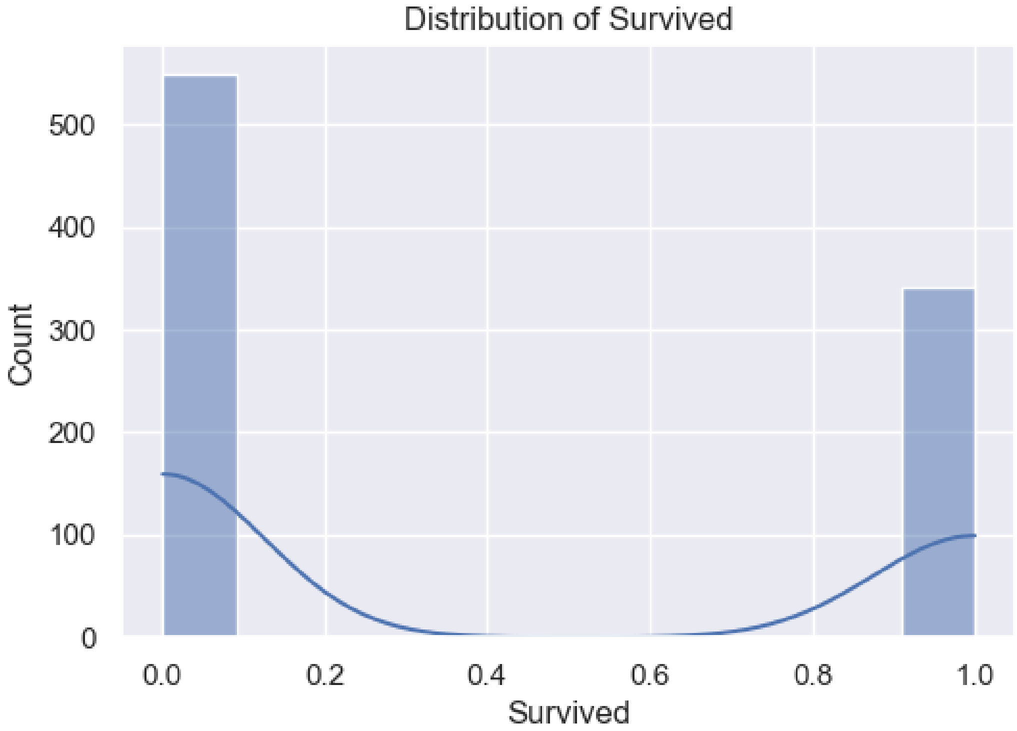

Log-Transformed Distribution of Parch
700
600
500
400
300
200
100
0
Count
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
Parch_log

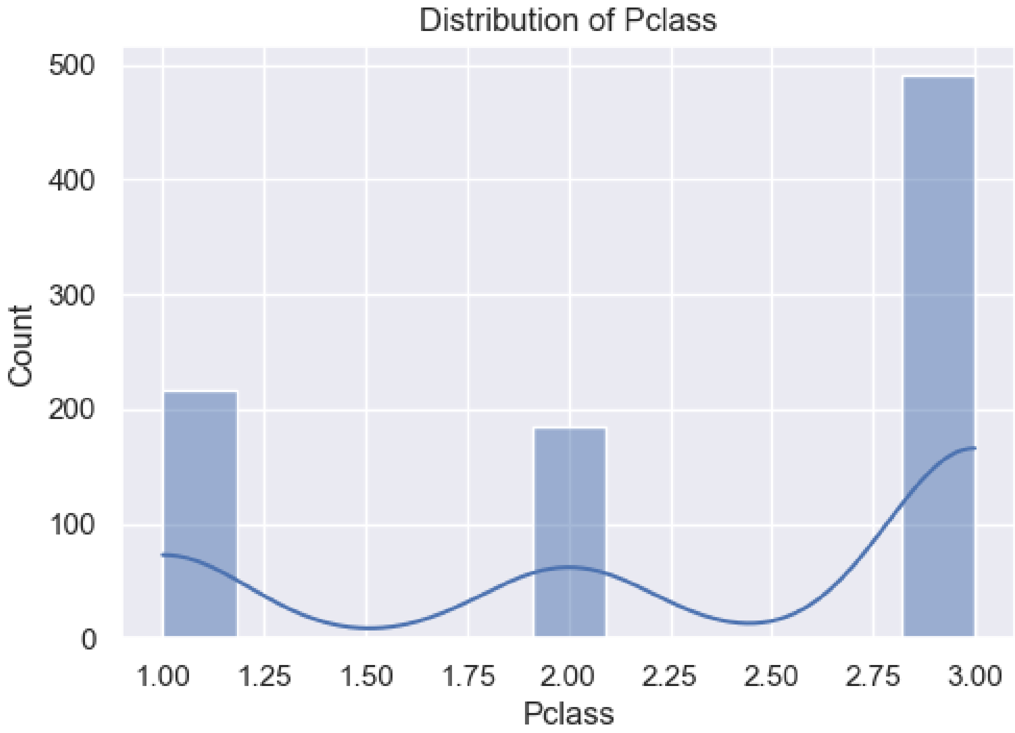

Original Distribution of Age
100
80
60
40
20
0
Count
0
10
20
30
40
50
60
70
80
Age

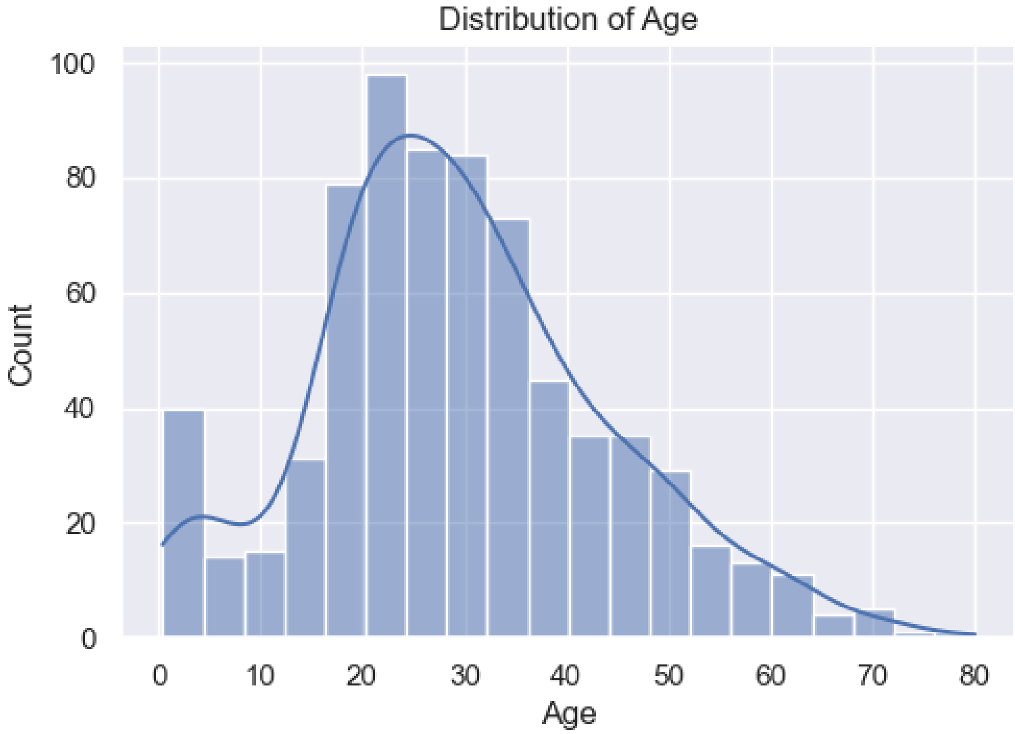


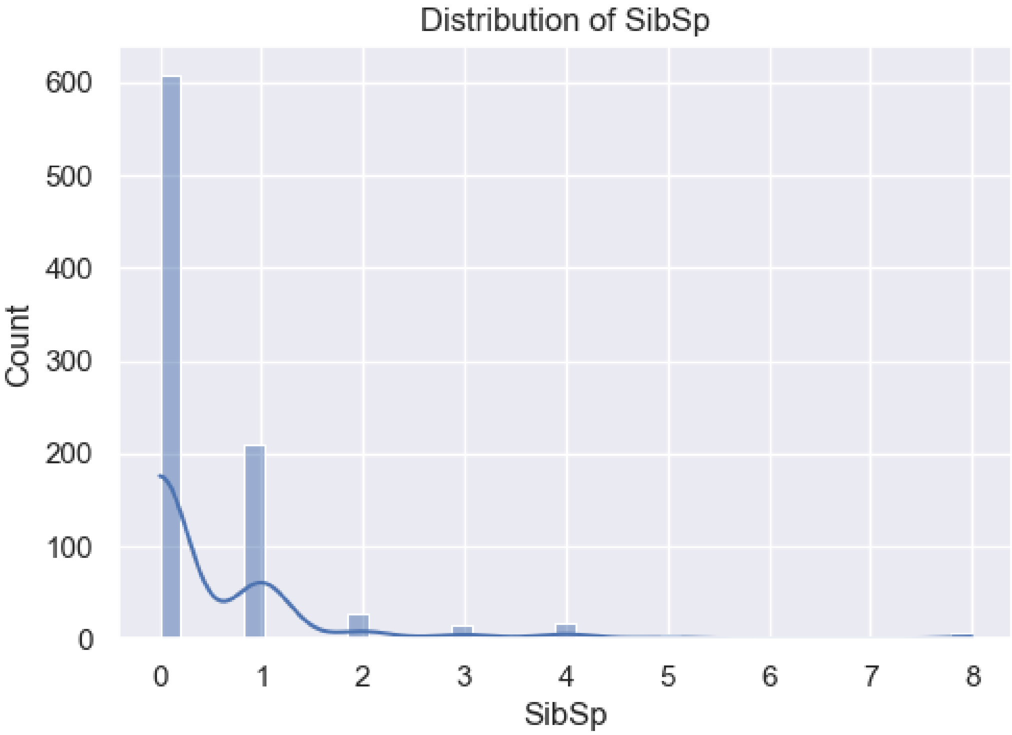
[394]:
```
for col in numeric_cols:
    plt.figure(figsize=(6,4))
    sns.histplot(df[col], kde=True)
    plt.title(f"Distribution of {col}")
    plt.xlabel(col)
    plt.ylabel("Count")
    plt.show()
```

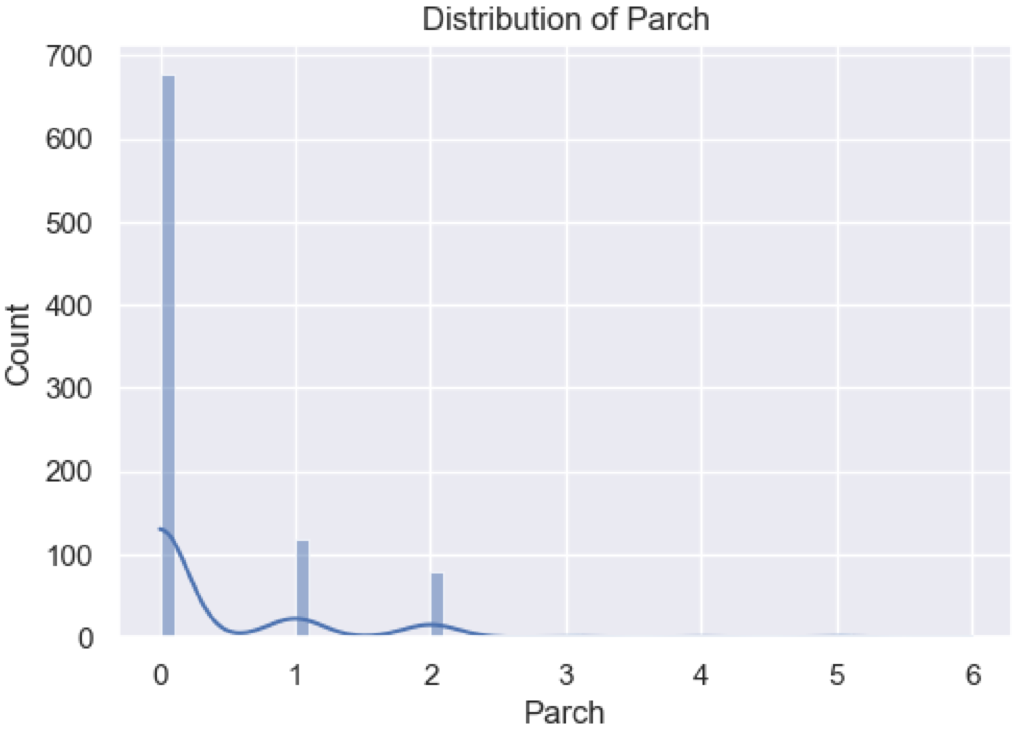

Distribution of PassengerId
Count
0
10
20
30
40
50
60
70
80
0
200
400
600
800
PassengerId

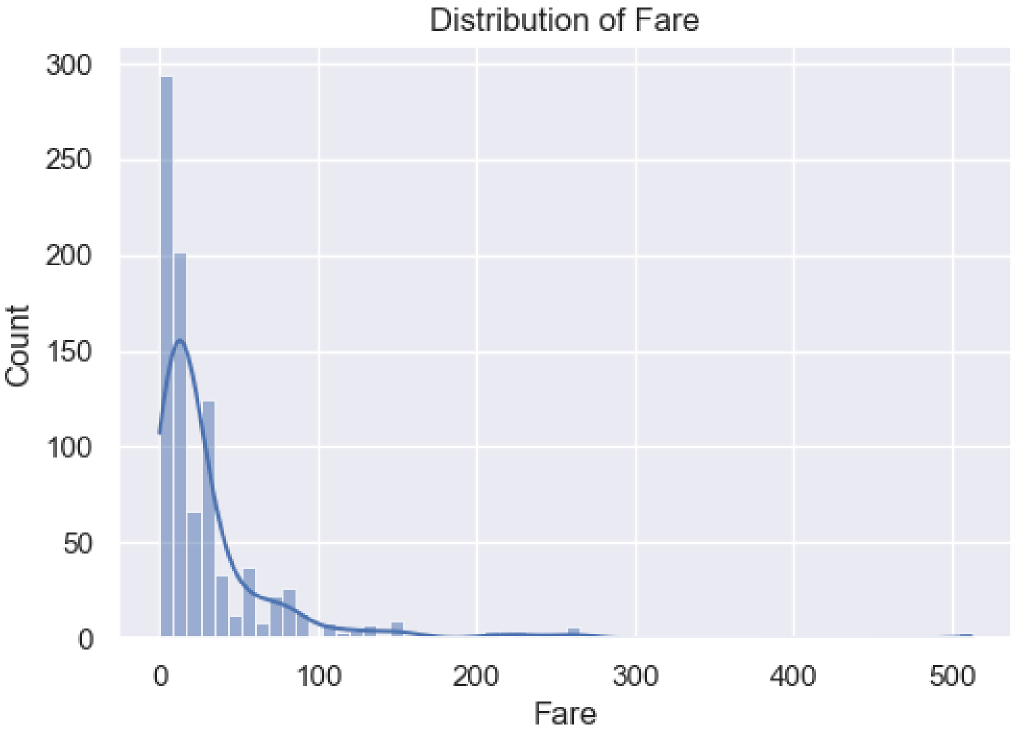

Distribution of Survived
Count
0
100
200
300
400
500
0.0
0.2
0.4
0.6
0.8
1.0
Survived


Distribution of Pclass
Count
0
100
200
300
400
500
1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75
3.00
Pclass


Distribution of Age
Count
0
20
40
60
80
100
0
10
20
30
40
50
60
70
80
Age


Distribution of SibSp
Count
600
500
400
300
200
100
0
0
1
2
3
4
5
6
7
8
SibSp


Distribution of Parch
Count
700
600
500
400
300
200
100
0
0
1
2
3
4
5
6
Parch


Distribution of Fare
Count
300
250
200
150
100
50
0
0
100
200
300
400
500
Fare