In [18]:

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

In [5]:

```
import pandas as pd
import zipfile

# Specify the zip file path
zip_file_path = '/content/titanic.zip'

# Specify the CSV file within the zip file you want to load
csv_file_name = 'train.csv'  # or 'gender_submission.csv', or 'test.csv'

# Open the zip file
with zipfile.ZipFile(zip_file_path, 'r') as zip_ref:
    # Extract the specific CSV file to a BytesIO object
    with zip_ref.open(csv_file_name) as file:
        # Read the CSV data into a pandas DataFrame
        df = pd.read_csv(file)

# Now you can work with the 'df' DataFrame
print(df.head())
```

```
   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  
```

In [6]:

```
# Information about columns
df.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
```

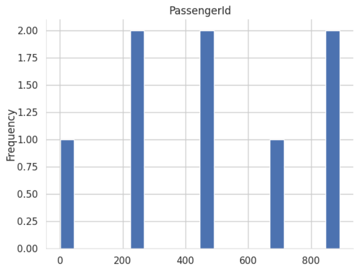
In [7]:

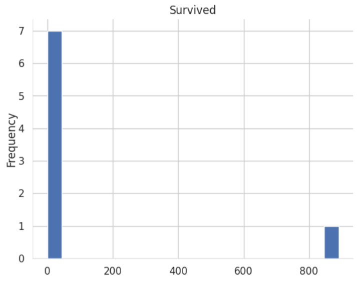
```
# Statistical summary
df.describe()
```

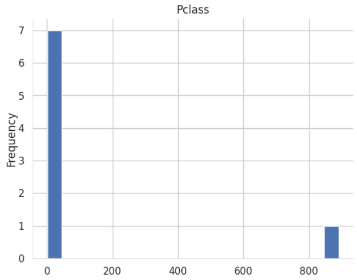
Out[7]:

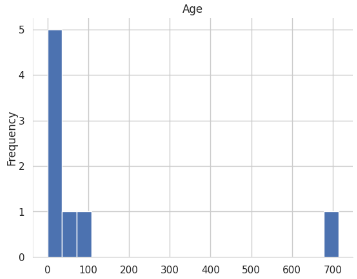
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	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

Distributions

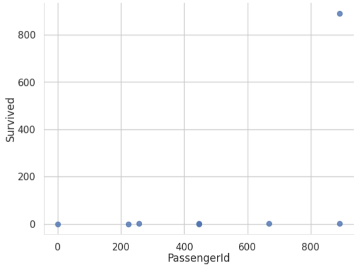


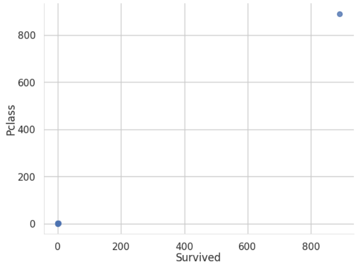


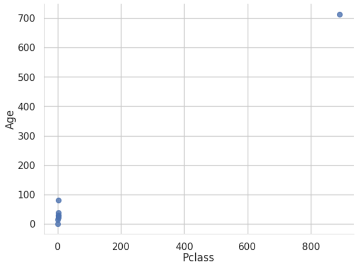


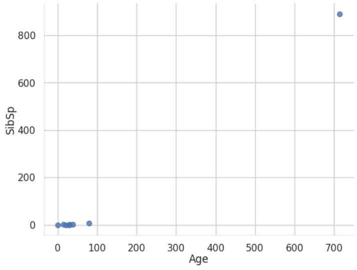


2-d distributions

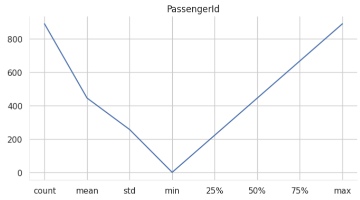


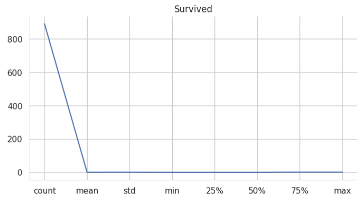


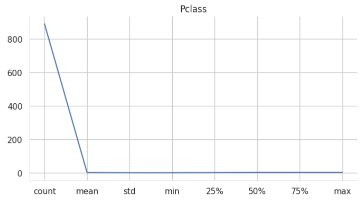


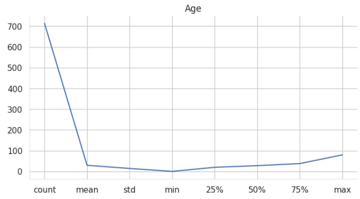


Values









In [8]:

```
# Check for missing values
df.isnull().sum()
```

Out[8]:

	0
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

In [9]:

```
# Value counts for categorical columns
df['Sex'].value_counts()
df['Embarked'].value_counts()
```

Out[9]:

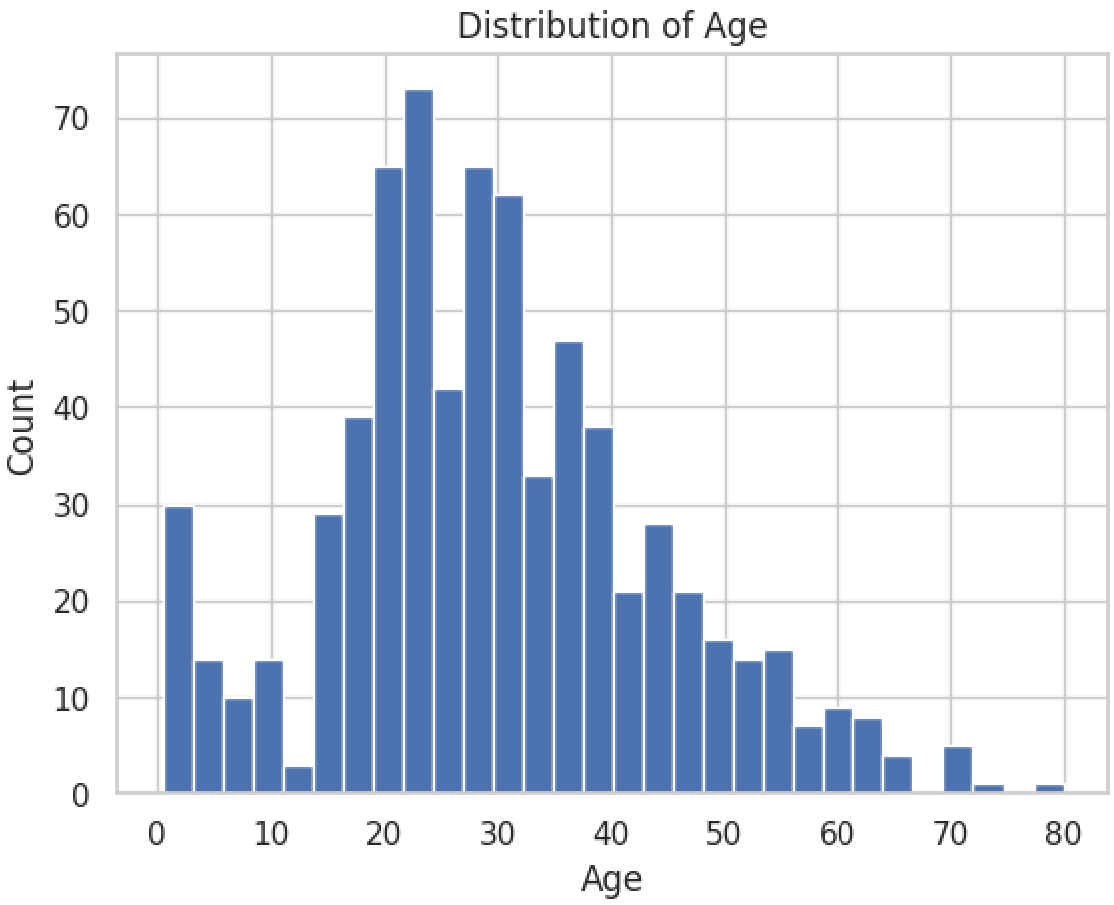
	count
Embarked	
S	644
C	168
Q	77

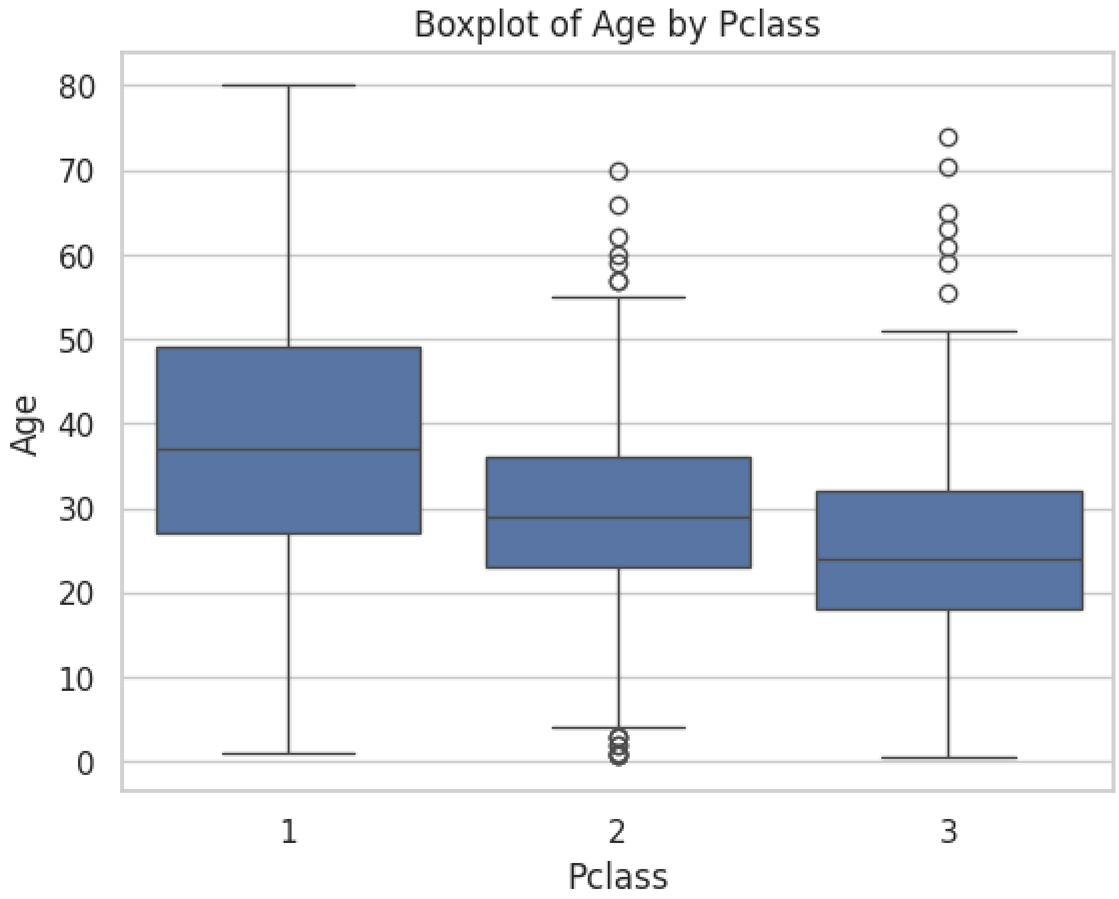
dtype: int64

In [10]:

```
# Histograms
df['Age'].hist(bins=30)
plt.title('Distribution of Age')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()

# Boxplots
sns.boxplot(x='Pclass', y='Age', data=df)
plt.title('Boxplot of Age by Pclass')
plt.show()
```

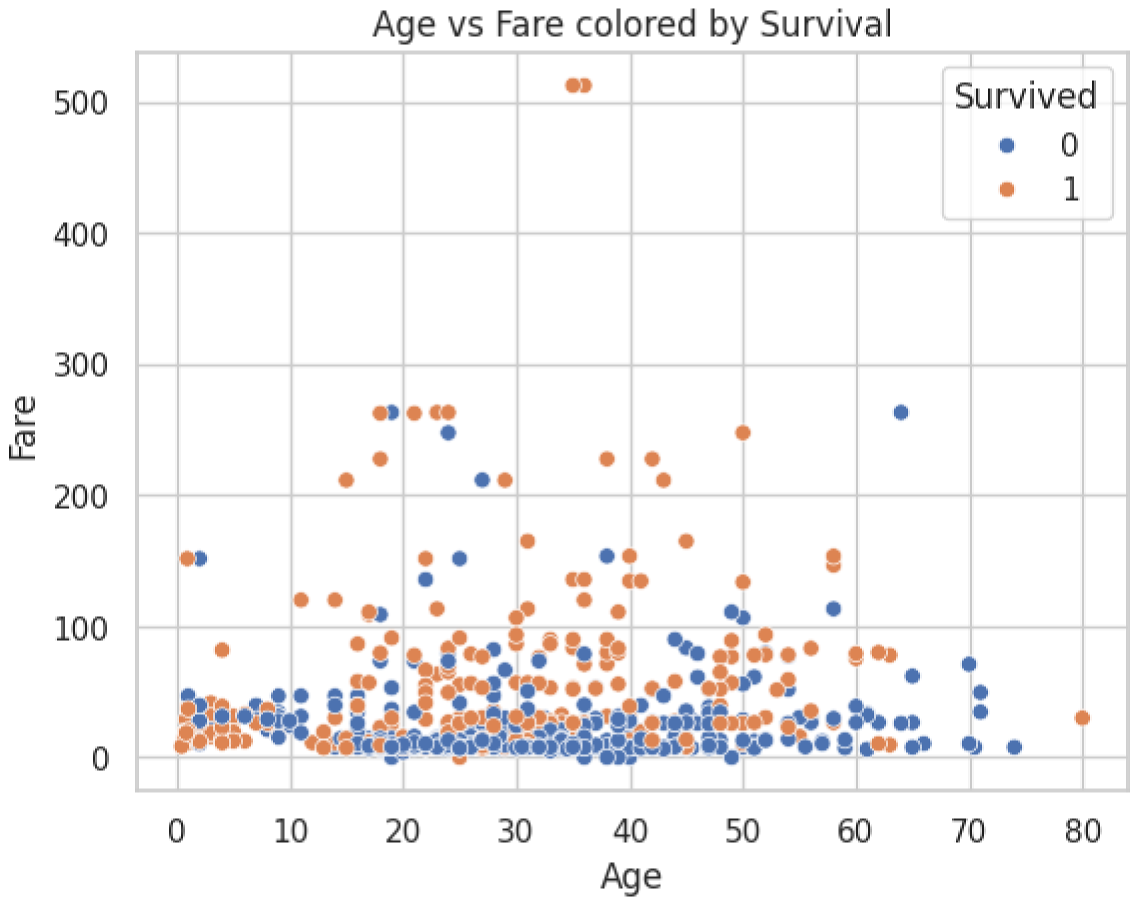


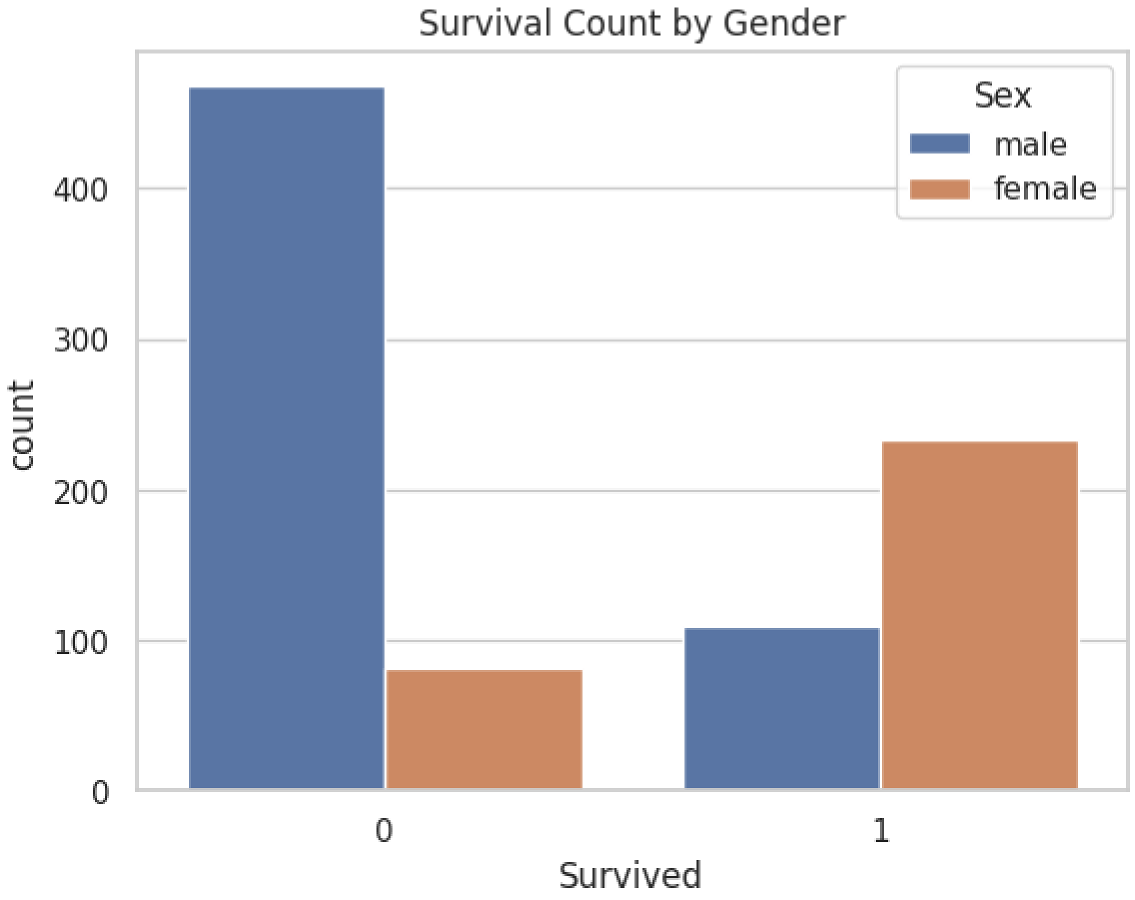


In [11]:

```
# Scatterplot
sns.scatterplot(x='Age', y='Fare', hue='Survived', data=df)
plt.title('Age vs Fare colored by Survival')
plt.show()

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



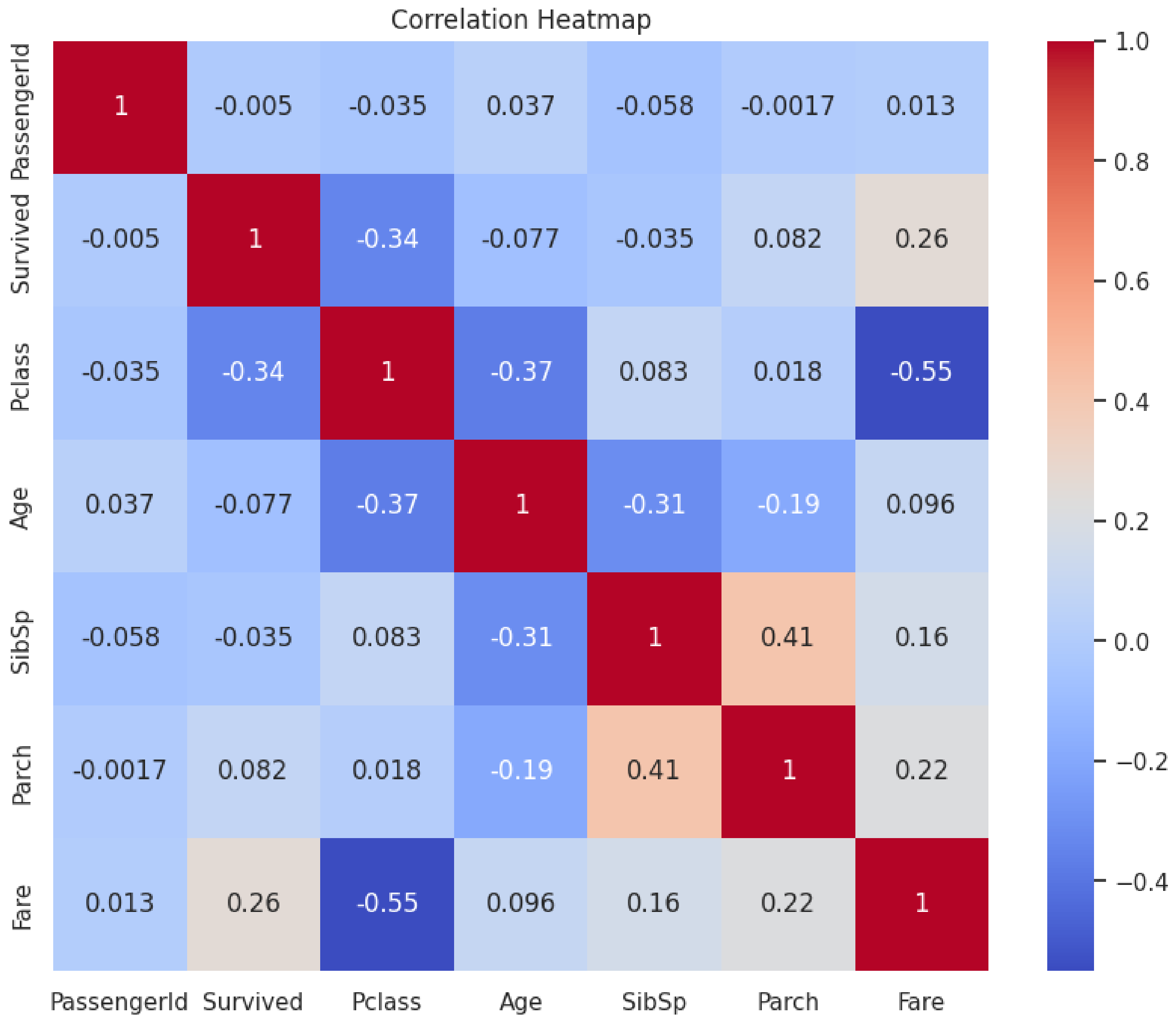


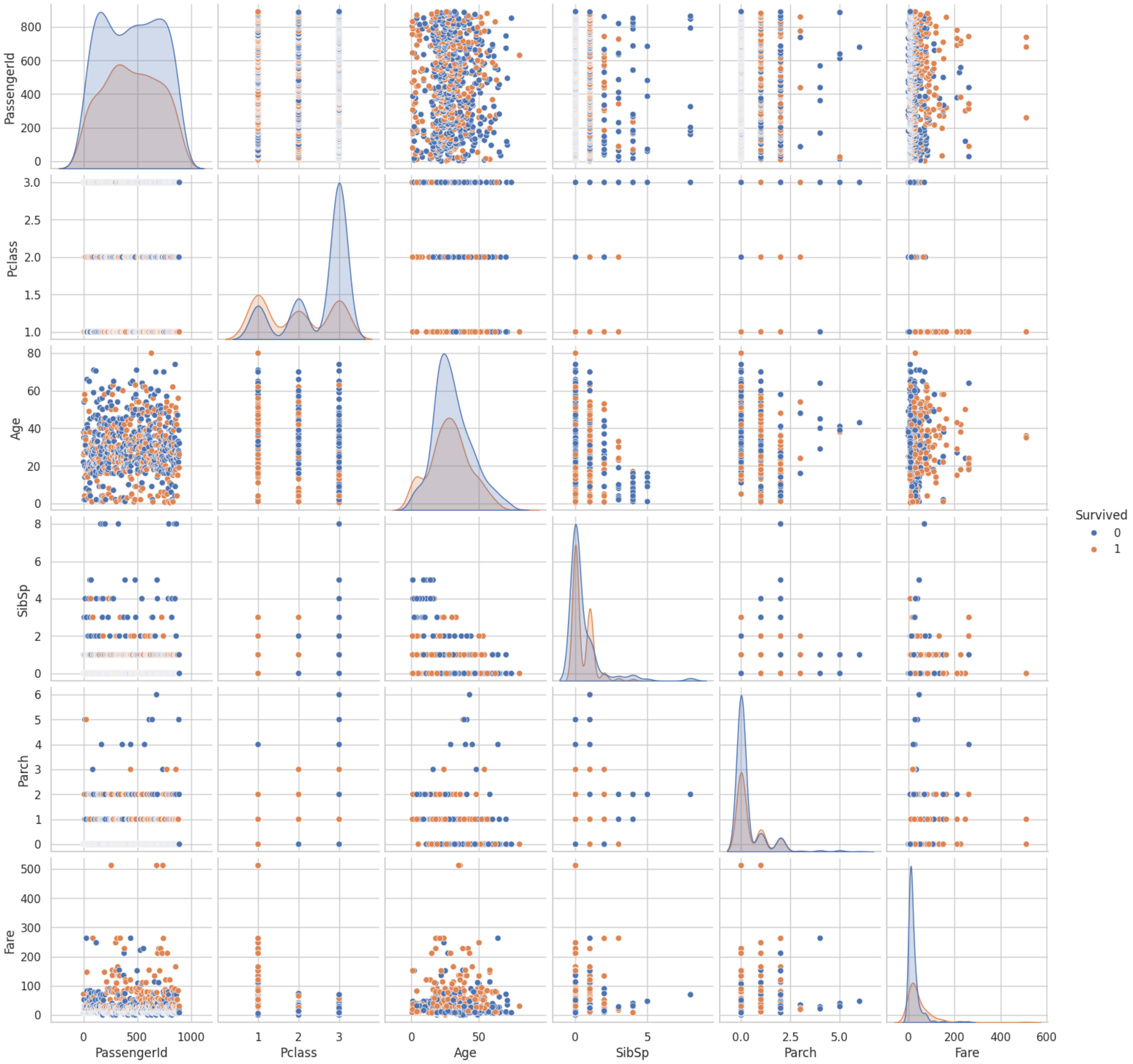
In [13]:

```
# Heatmap of correlations
plt.figure(figsize=(10,8))
```

```
# Select only numeric features for correlation
numeric_df = df.select_dtypes(include=np.number)
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()

# Pairplot
# Select only numeric features for pairplot
sns.pairplot(numeric_df, hue='Survived') # Assuming 'Survived' is numeric
plt.show()
```




PassengerId
Pclass
Age
SibSp
Parch
Fare
800
600
400
200
0
3.0
2.5
2.0
1.5
1.0
80
60
40
20
8
6
4
2
6
5
4
3
1
500
400
300
200
100
0
500
1000
1
2
3
50
5
0.0
2.5
5.0
600
Survived
0
1