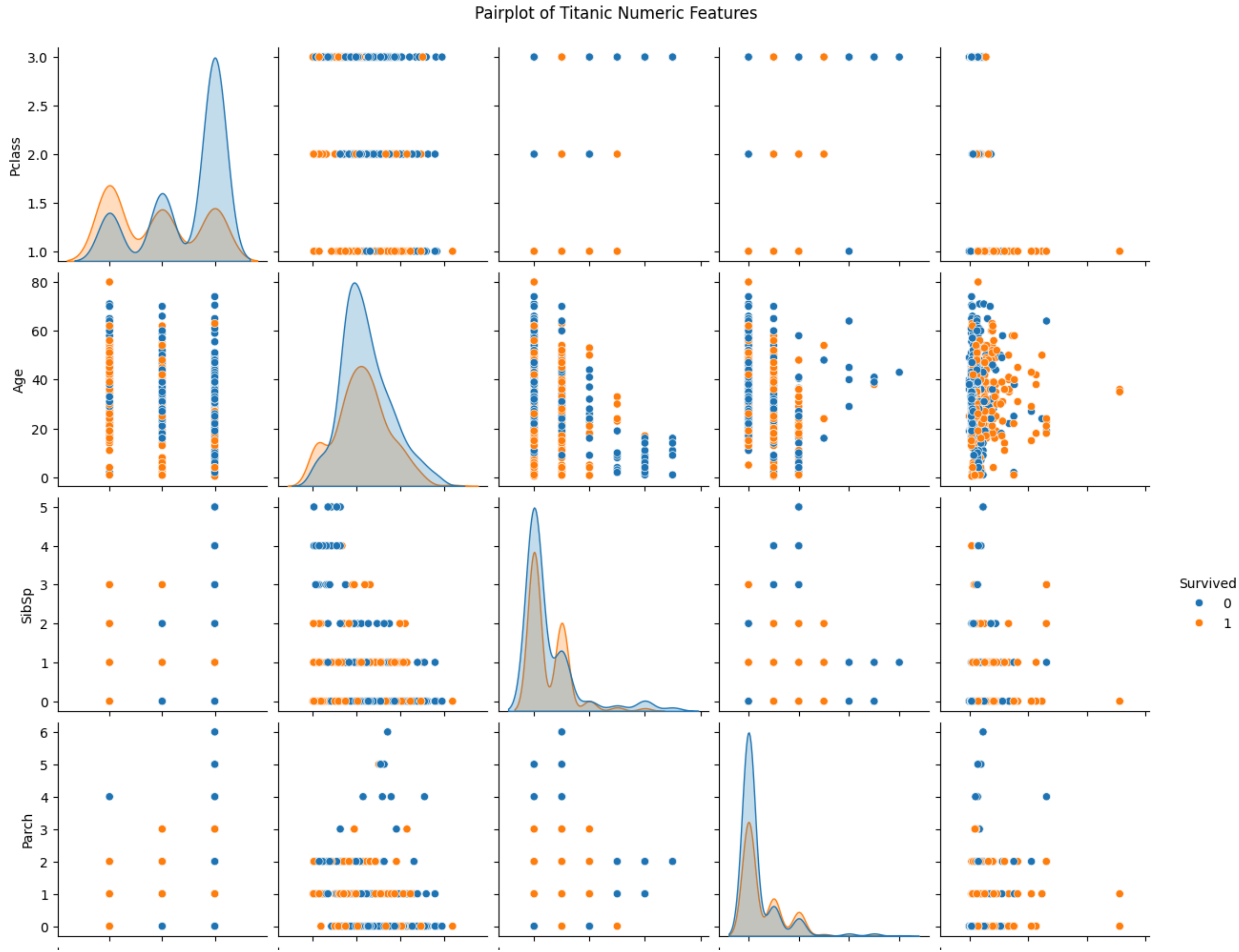
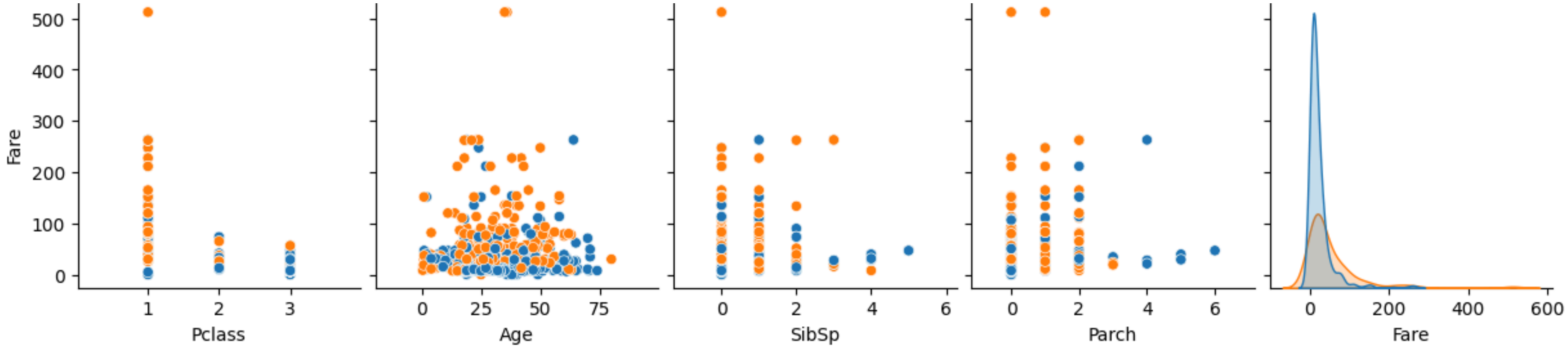
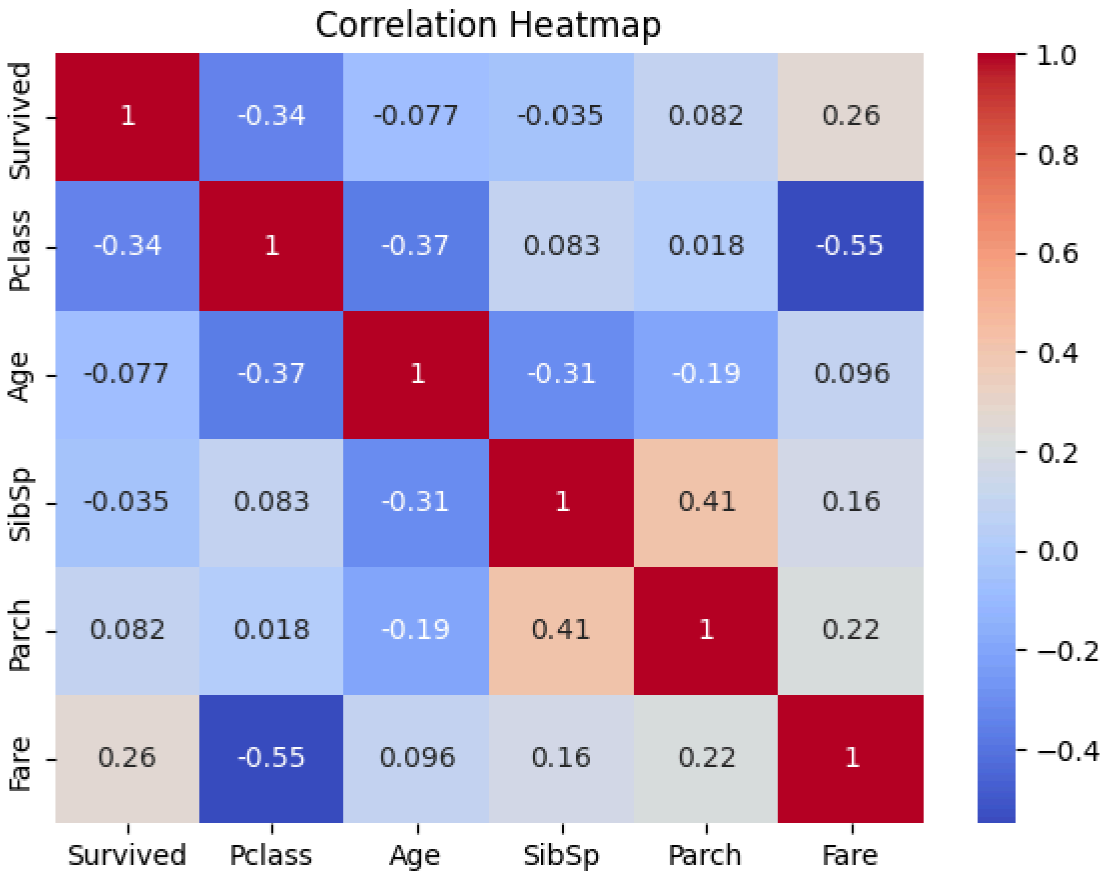
In [6]:
```
sns.pairplot(df[['Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']].dropna(),
             hue='Survived', diag_kind='kde')
plt.suptitle("Pairplot of Titanic Numeric Features", y=1.02)
plt.show()

plt.figure(figsize=(7,5))
sns.heatmap(df[['Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']].corr(),
            annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```


Pairplot of Titanic Numeric Features
Pclass
Age
SibSp
Parch
3.0
2.5
2.0
1.5
1.0
80
60
40
20
0
5
4
3
2
1
0
6
Survived
0
1





Pairplot Observations:

1. Younger passengers had a higher survival rate compared to older ones.
2. Passengers in Pclass 1 had a better survival rate than those in lower classes.
3. Higher fare values are associated with higher survival chances.

Heatmap Observations:

1. Survival has a negative correlation with Pclass (-0.34), meaning higher classes survived more.
2. Fare and Pclass are strongly negatively correlated (-0.55), higher class = higher fare.
3. Age does not have a strong correlation with survival.

In [5]:
```
# ---------------- a) Load dataset & basic info ----------------
df=  pd.read_csv("train.csv")
                              # make sure the file is in your working directory

print("=== Data Info ===")
df.info()

print("\n=== Summary Statistics (numeric) ===")
print(df.describe())

print("\n=== Summary Statistics (all columns) ===")
print(df.describe(include='all'))

print("\n=== Value Counts for key categorical columns ===")
categorical_cols = ['Sex', 'Pclass', 'Embarked', 'Survived']  # adjust if your column names differ
for col in categorical_cols:
    if col in df.columns:
        print(f"\nValue counts for {col}:")
        print(df[col].value_counts(dropna=False))
```

```
=== 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

=== Summary Statistics (numeric) ===
       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  
```

```
=== Summary Statistics (all columns) ===
        PassengerId    Survived      Pclass                 Name   Sex  \
count    891.000000  891.000000  891.000000                  891   891   
unique          NaN         NaN         NaN                  891     2   
top             NaN         NaN         NaN  Dooley, Mr. Patrick  male   
freq            NaN         NaN         NaN                    1   577   
mean     446.000000    0.383838    2.308642                  NaN   NaN   
std      257.353842    0.486592    0.836071                  NaN   NaN   
min        1.000000    0.000000    1.000000                  NaN   NaN   
25%      223.500000    0.000000    2.000000                  NaN   NaN   
50%      446.000000    0.000000    3.000000                  NaN   NaN   
75%      668.500000    1.000000    3.000000                  NaN   NaN   
max      891.000000    1.000000    3.000000                  NaN   NaN   

               Age       SibSp       Parch  Ticket        Fare Cabin Embarked  
count   714.000000  891.000000  891.000000     891  891.000000   204      889  
unique         NaN         NaN         NaN     681         NaN   147        3  
top            NaN         NaN         NaN  347082         NaN    G6        S  
freq           NaN         NaN         NaN       7         NaN     4      644  
mean     29.699118    0.523008    0.381594     NaN   32.204208   NaN      NaN  
std      14.526497    1.102743    0.806057     NaN   49.693429   NaN      NaN  
min       0.420000    0.000000    0.000000     NaN    0.000000   NaN      NaN  
25%      20.125000    0.000000    0.000000     NaN    7.910400   NaN      NaN  
50%      28.000000    0.000000    0.000000     NaN   14.454200   NaN      NaN  
75%      38.000000    1.000000    0.000000     NaN   31.000000   NaN      NaN  
max      80.000000    8.000000    6.000000     NaN  512.329200   NaN      NaN  

=== Value Counts for key categorical columns ===

Value counts for Sex:
Sex
male      577
female    314
Name: count, dtype: int64

Value counts for Pclass:
Pclass
3    491
1    216
2    184
```

```
Name: count, dtype: int64

Value counts for Embarked:
Embarked
S      644
C      168
Q       77
NaN      2
Name: count, dtype: int64

Value counts for Survived:
Survived
0    549
1    342
Name: count, dtype: int64
```

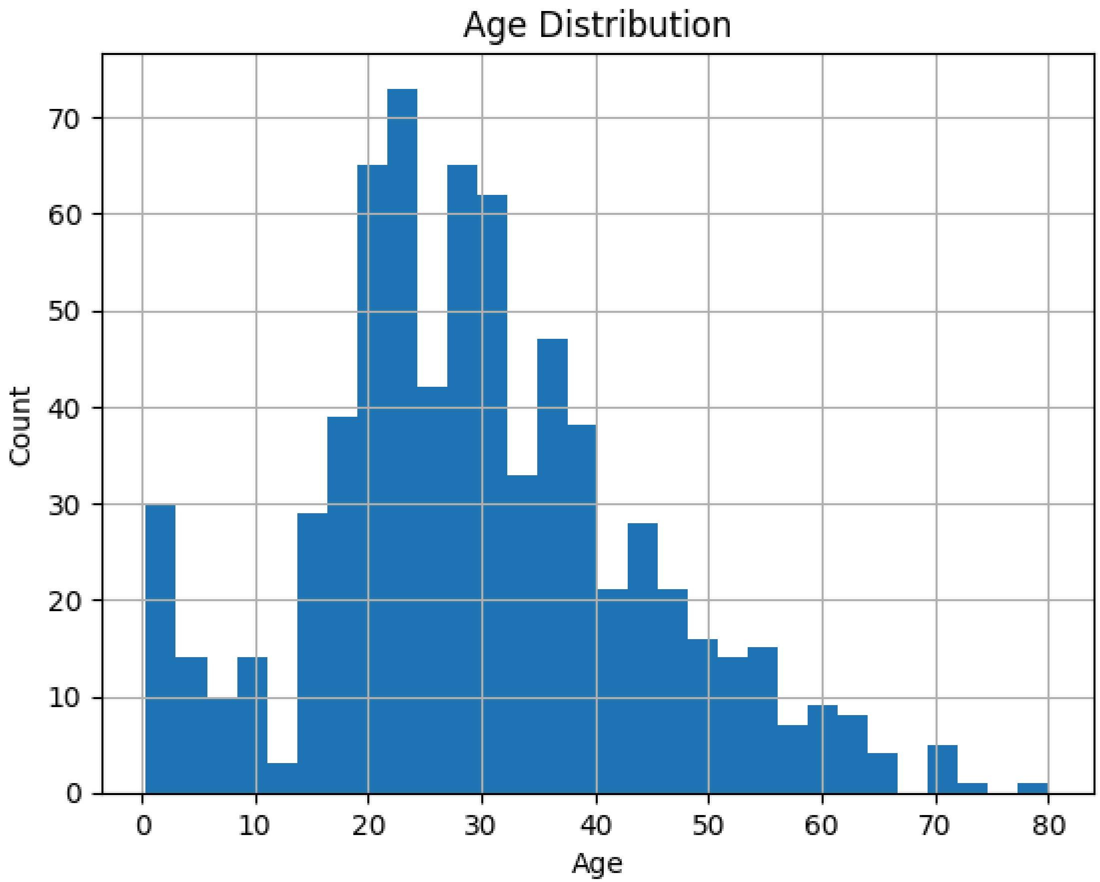
In [7]:

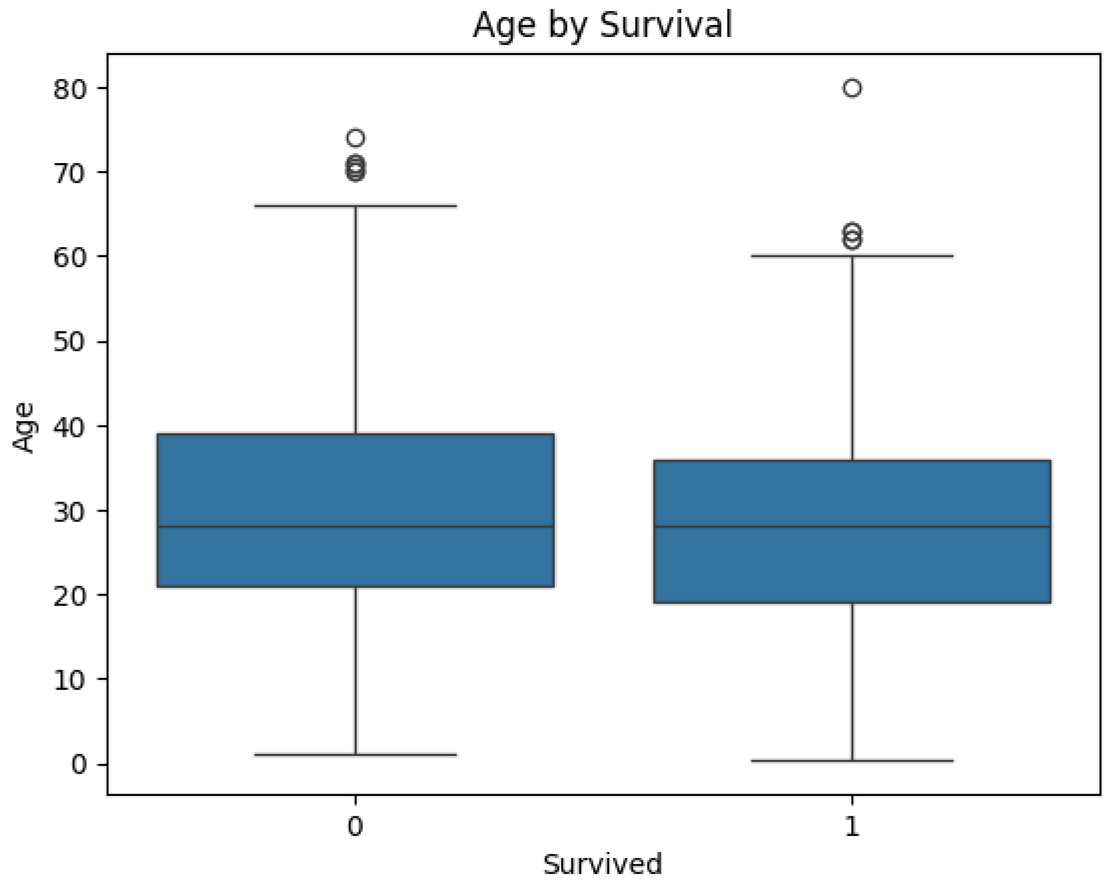
```
# ---------------- c) Identify relationships and trends ----------------
print("\n=== Survival rate by Sex ===")
print(df.groupby('Sex')['Survived'].mean())

print("\n=== Survival rate by Pclass ===")
print(df.groupby('Pclass')['Survived'].mean())

print("\n=== Survival rate by Embarked ===")
print(df.groupby('Embarked')['Survived'].mean())
```

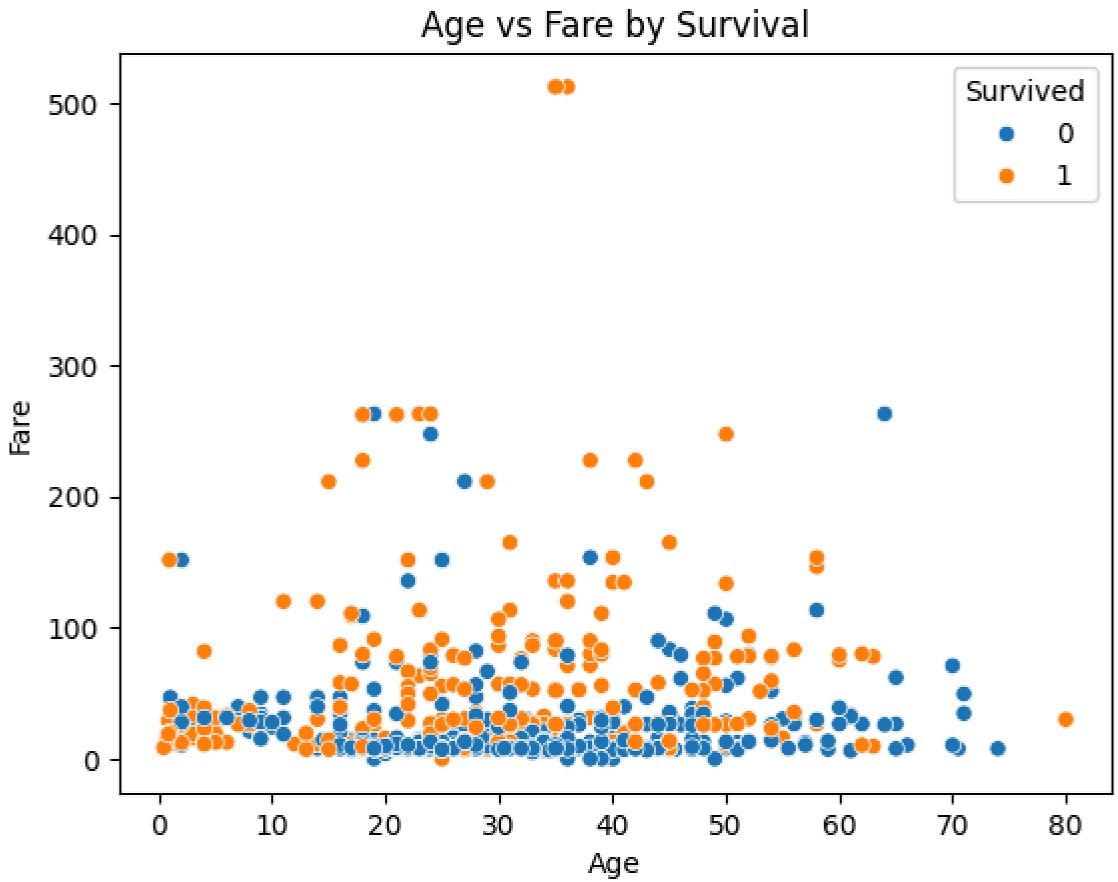
```
=== Survival rate by Sex ===
Sex
female    0.742038
male      0.188908
Name: Survived, dtype: float64

=== Survival rate by Pclass ===
Pclass
1    0.629630
2    0.472826
3    0.242363
Name: Survived, dtype: float64

=== Survival rate by Embarked ===
Embarked
C    0.553571
Q    0.389610
S    0.336957
Name: Survived, dtype: float64
```

In [8]:

```
# ---------------- d) Histograms, Boxplots, Scatterplots ----------------
# Histogram
df['Age'].hist(bins=30)
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Count")
plt.show()

# Boxplot: Age by Survival
sns.boxplot(x='Survived', y='Age', data=df)
plt.title("Age by Survival")
plt.show()

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


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


Age by Survival
80
70
60
50
40
30
20
10
0
Age
0
1
Survived



e) Observations: Extra Plots

1. Most passengers were between **20–40 years old**.
2. Survivors tended to have paid **higher fares** on average.

---

7. Summary of Findings

- Higher class passengers (**Pclass 1**) had better survival chances.

- **Females** and **younger passengers** had a higher survival probability.
- **Fare** is positively related to survival — possibly indicating access to better cabins/lifeboats.
- **SibSp** and **Parch** have weak relationships with survival, but traveling with small family groups might have helped survival chances.

In [3]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```