In [15]:
```
# Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

pd.set_option("display.max_columns", None)
sns.set(style="whitegrid")
```

In [9]:
```
# Load the Dataset
df=pd.read_csv('/content/train.csv')
```

In [23]:
```
# Structure of dataset
print(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
None
```

In [ ]:
```
# Statistical summary
df.describe()
```

Out[ ]:

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

In [ ]:
```
# Missing values
df.isnull().sum()
```

Out[ ]:

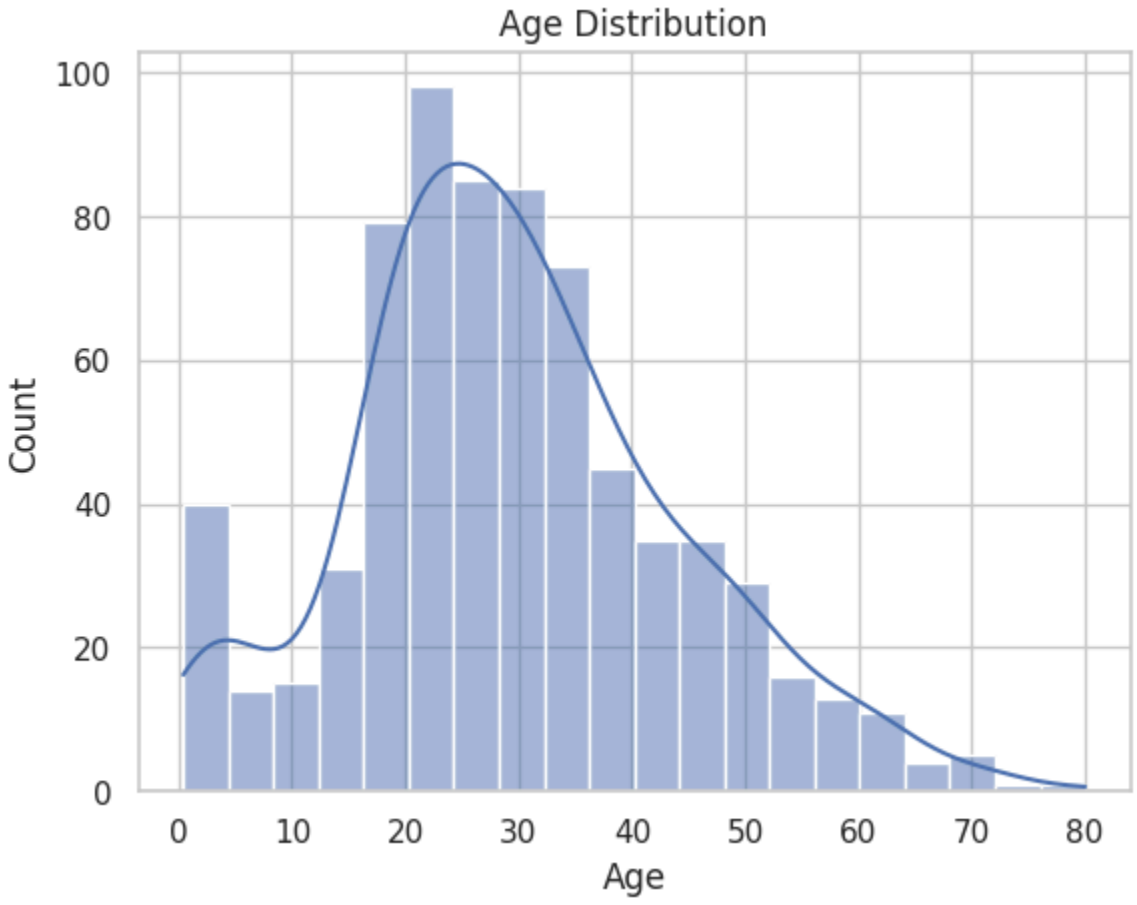
	0
PassengerId	0
Survived	0
Pclass	0
Name	0
Sex	0
Age	177
SibSp	0
Parch	0
Ticket	0
Fare	0
Cabin	687
Embarked	2

dtype: int64

In [24]:
```
# Value counts of categorical columns
for col in ['Sex', 'Embarked', 'Pclass']:
    print(df[col].value_counts())
    print("------")
```

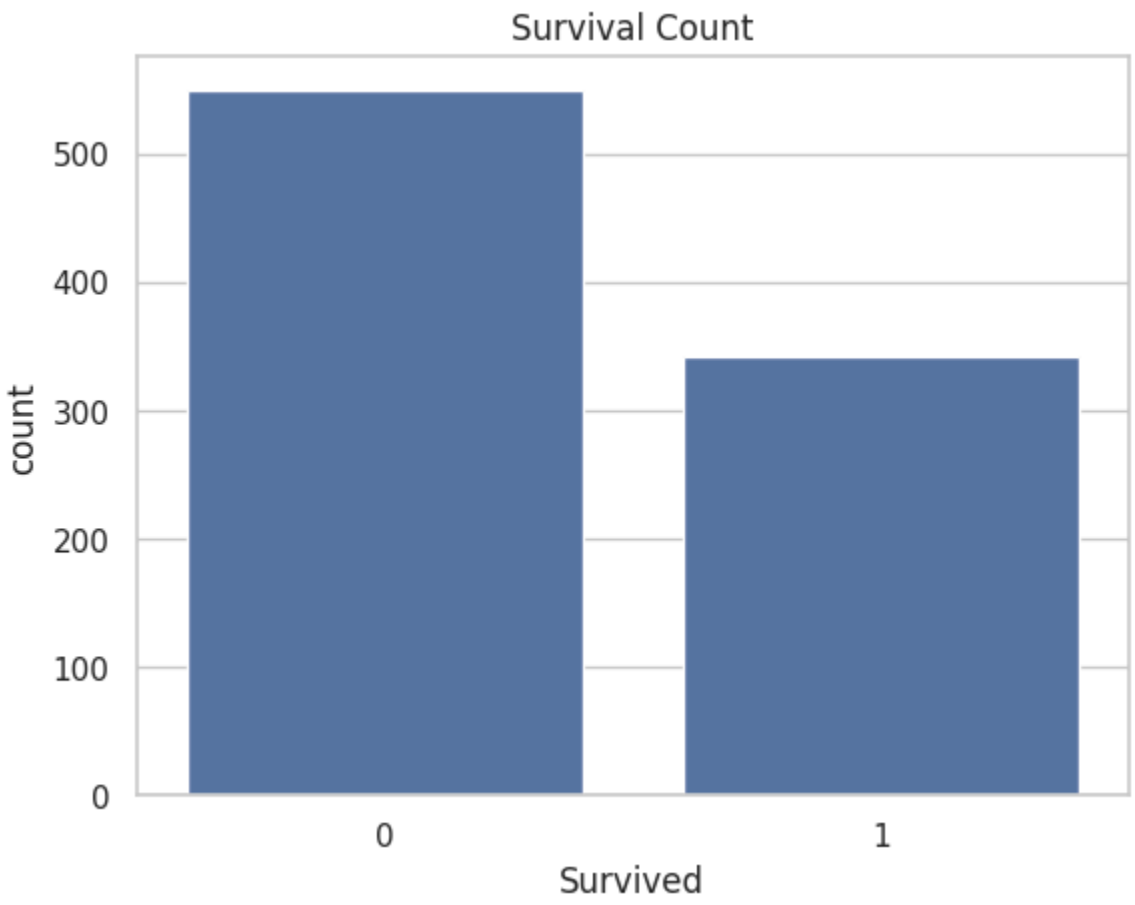
```
Sex
male      577
female    314
Name: count, dtype: int64
------
Embarked
S    644
C    168
Q     77
Name: count, dtype: int64
------
Pclass
3    491
1    216
2    184
Name: count, dtype: int64
------
```

In [16]:
```
# Histogram of Age
sns.histplot(df['Age'].dropna(), kde=True)
plt.title("Age Distribution")
plt.show()
```

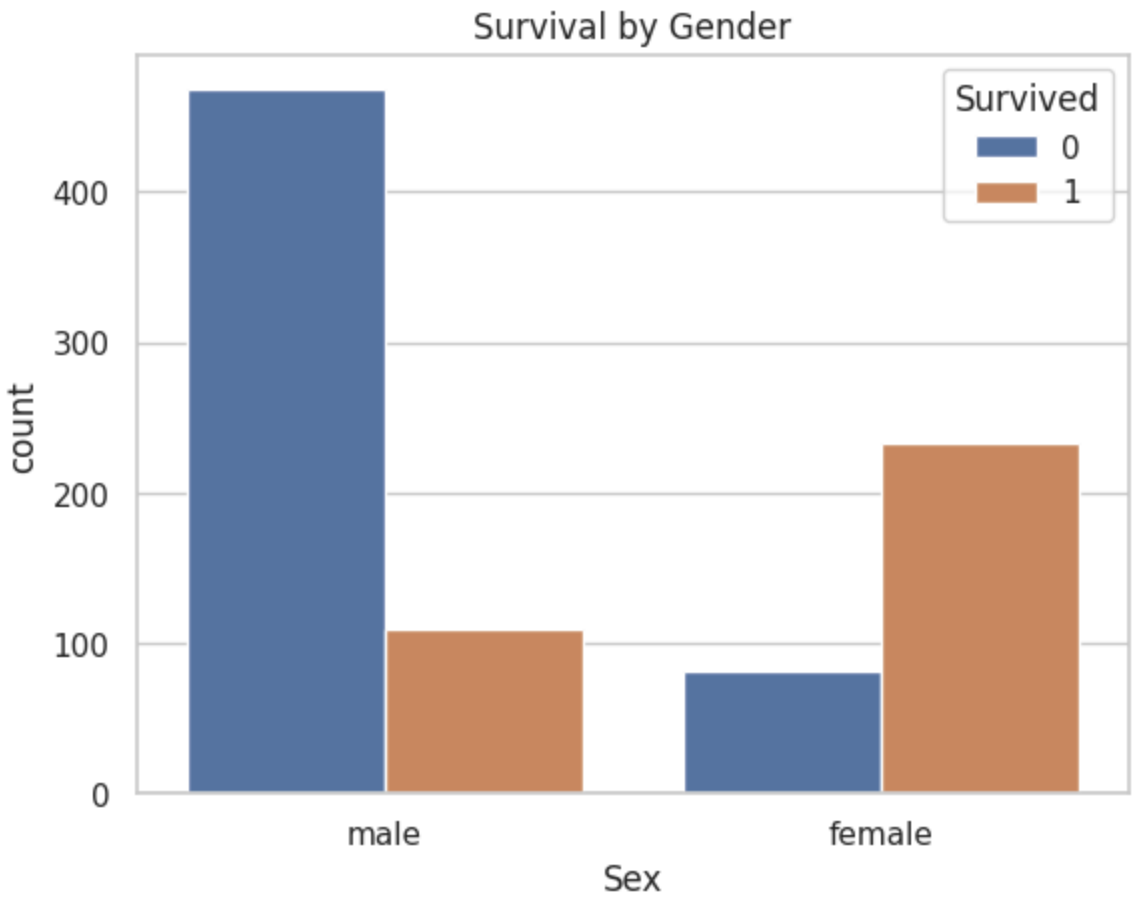


In [17]:
```
# Countplot of Survived
sns.countplot(x='Survived', data=df)
plt.title("Survival Count")
```

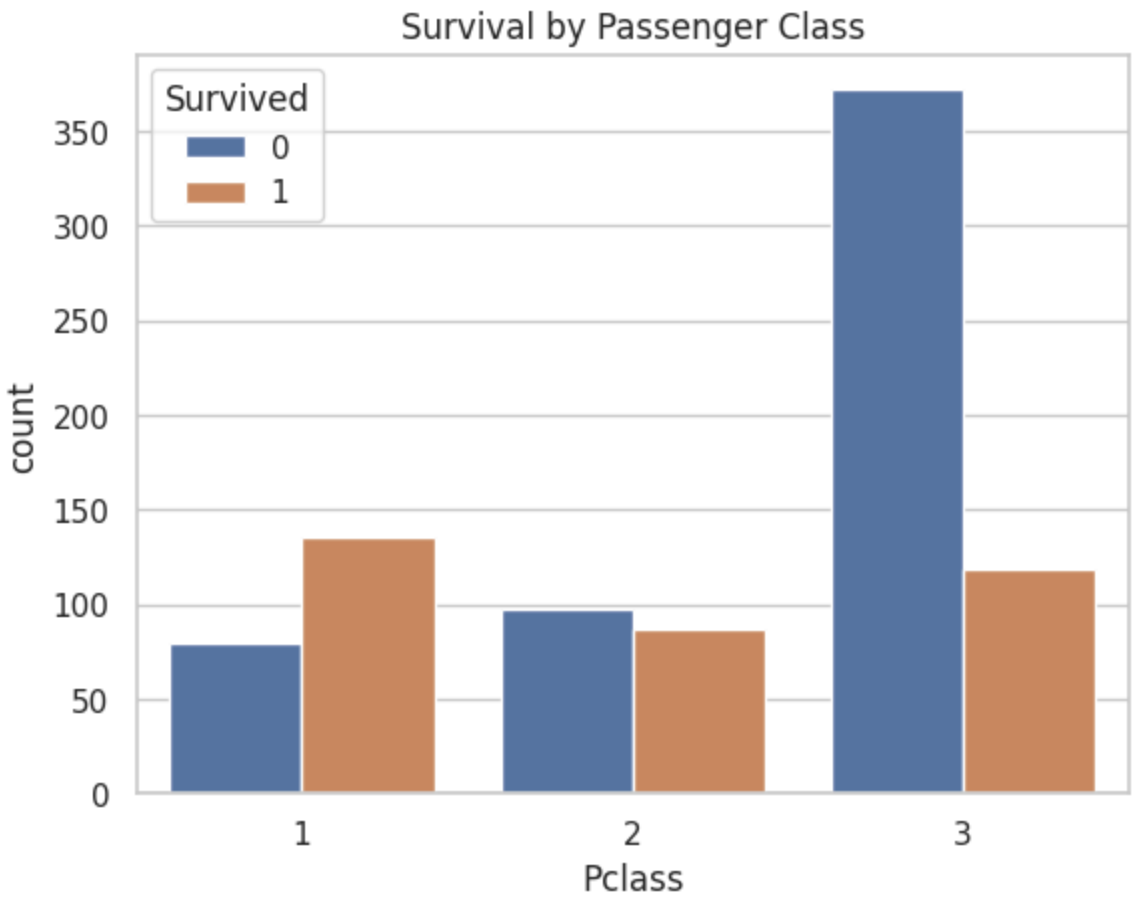
```
plt.show()
```



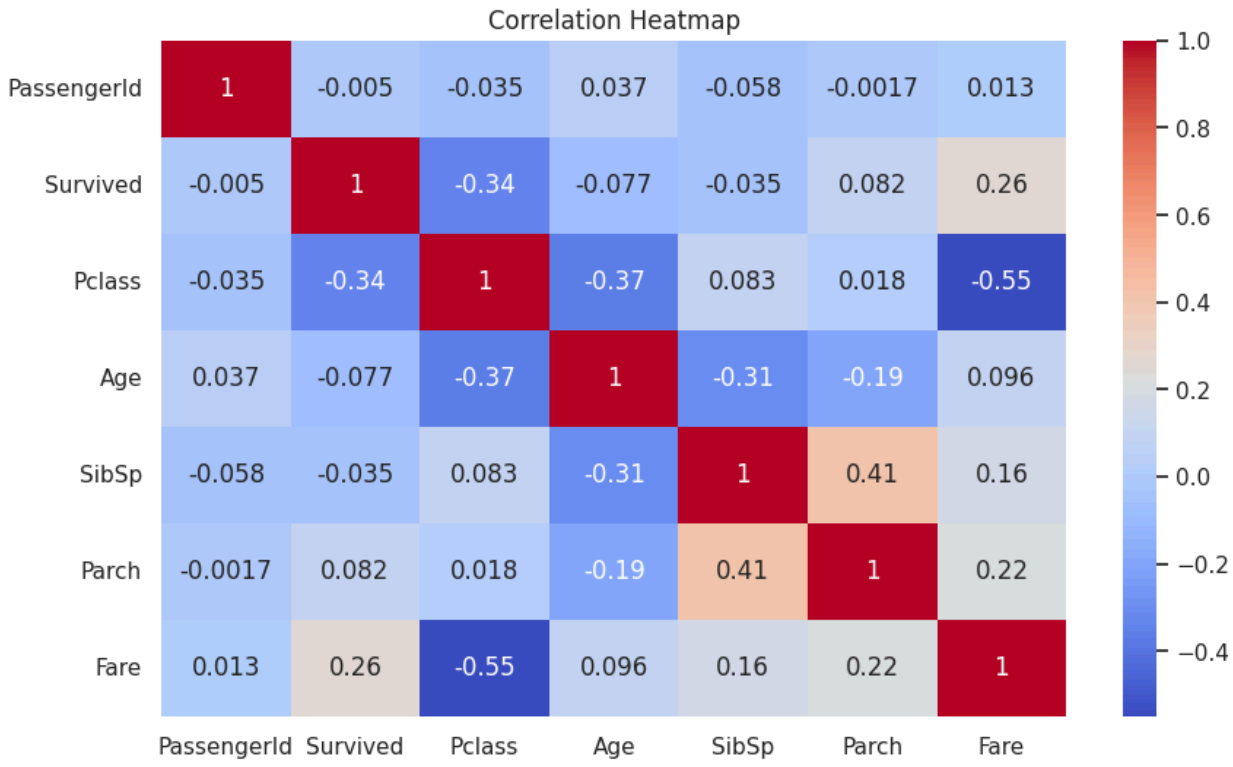
In [18]:
```
# Survival rate by gender
sns.countplot(x='Sex', hue='Survived', data=df)
plt.title("Survival by Gender")
plt.show()
```



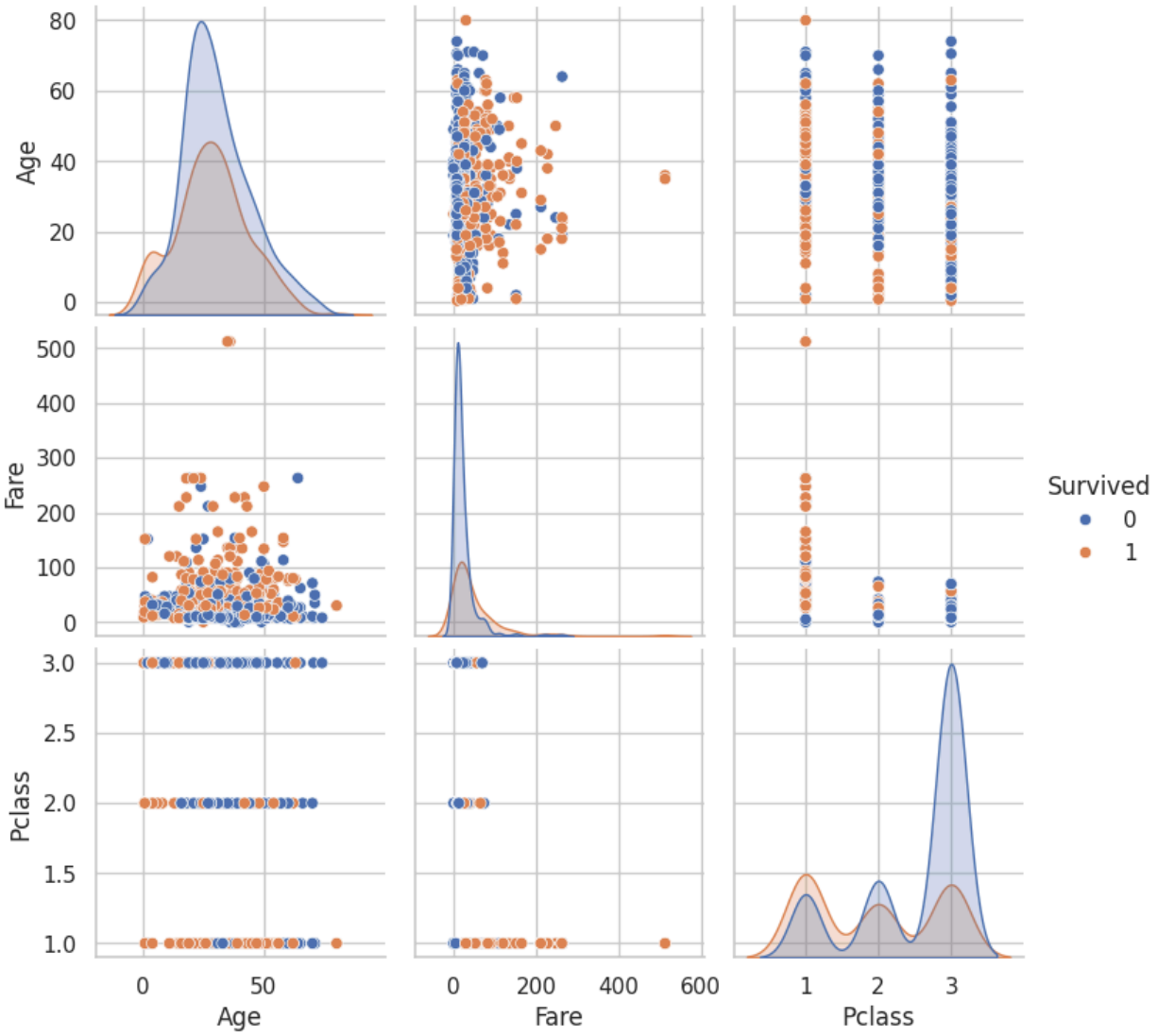
In [19]:
```
# Survival rate by Pclass
sns.countplot(x='Pclass', hue='Survived', data=df)
plt.title("Survival by Passenger Class")
plt.show()
```



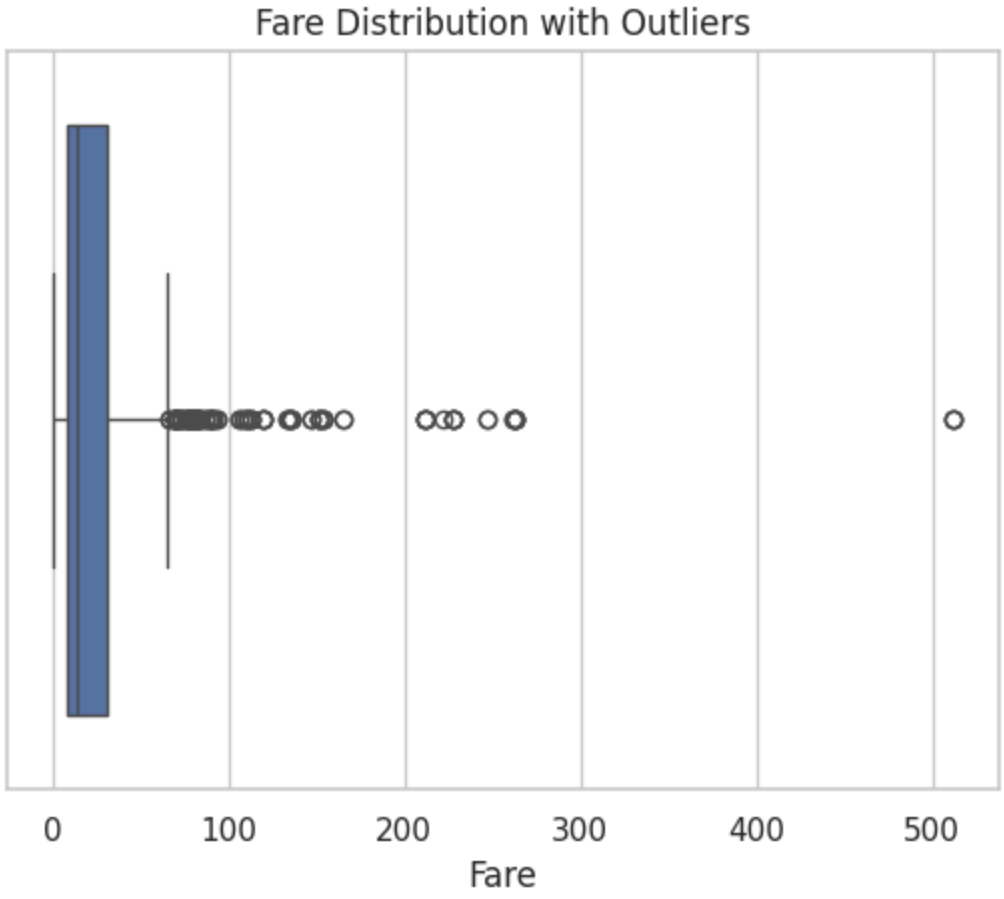
In [20]:
```
# Heatmap of correlation
plt.figure(figsize=(10,6))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap="coolwarm")
plt.title("Correlation Heatmap")
plt.show()
```



In [21]:
```
# Pairplot (only selected features to avoid clutter)
sns.pairplot(df[['Survived', 'Age', 'Fare', 'Pclass']], hue='Survived')
plt.show()
```



In [22]:
```
# Boxplot of Fare
sns.boxplot(x=df['Fare'])
plt.title("Fare Distribution with Outliers")
plt.show()
```


Fare Distribution with Outliers
0
100
200
300
400
500
Fare