In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [ ]:
```
file_path = r"Task 5\Titanic\train.csv"
train = pd.read_csv(file_path)
```

In [6]:
```
train.head()
```

Out[6]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500

In [7]:
```
train.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 [8]:
```
print(train.isnull().sum())
```

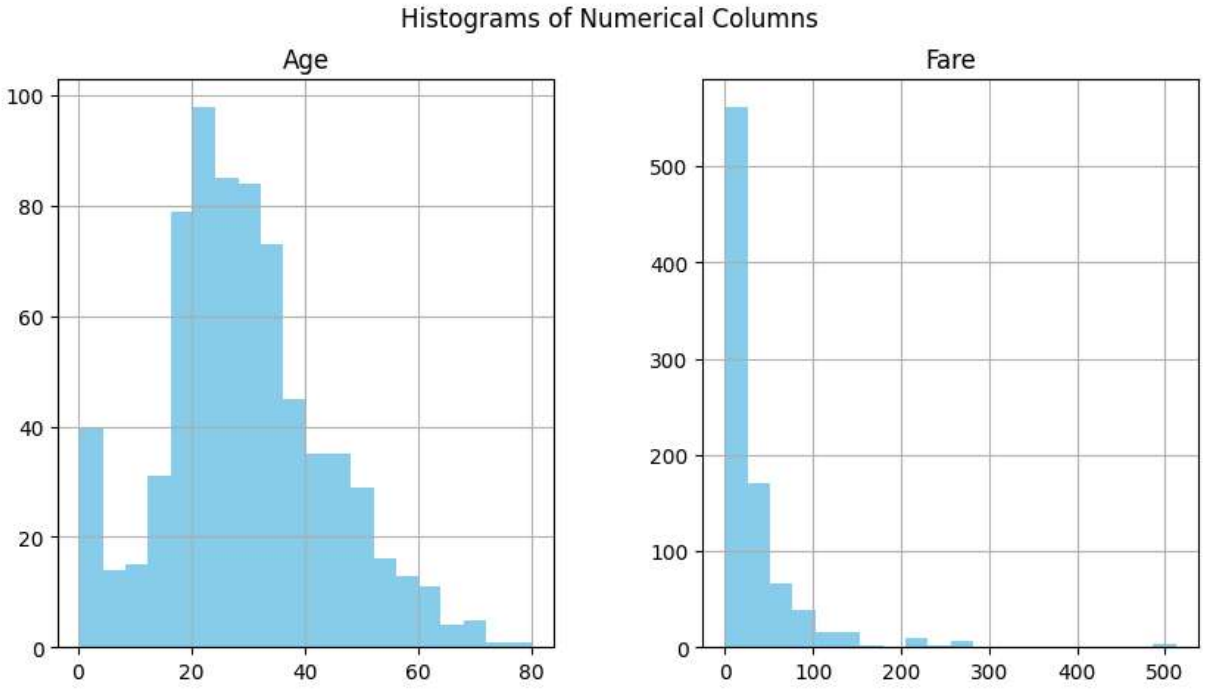
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

In [9]:
```
display(train.describe(include='all'))
```

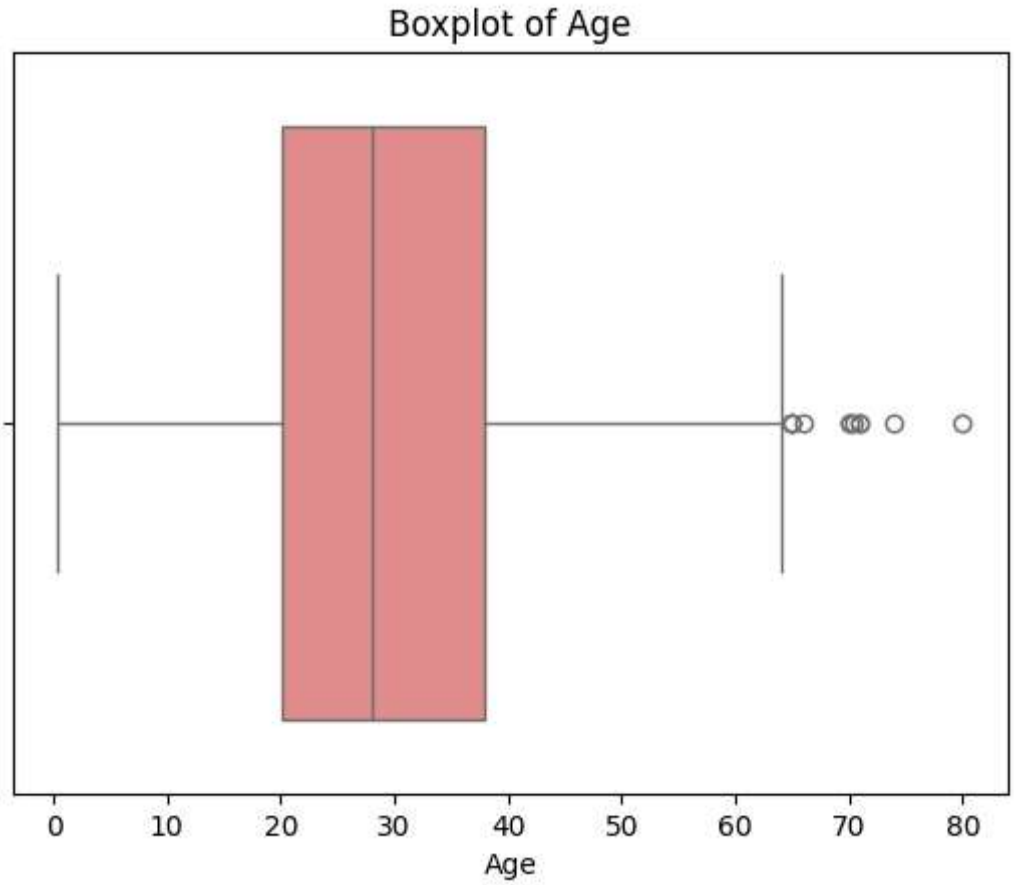
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Pa
count	891.000000	891.000000	891.000000	891	891	714.000000	891.000000	891.000
unique	NaN	NaN	NaN	891	2	NaN	NaN	N
top	NaN	NaN	NaN	Braund, Mr. Owen Harris	male	NaN	NaN	N
freq	NaN	NaN	NaN	1	577	NaN	NaN	N
mean	446.000000	0.383838	2.308642	NaN	NaN	29.699118	0.523008	0.381
std	257.353842	0.486592	0.836071	NaN	NaN	14.526497	1.102743	0.806
min	1.000000	0.000000	1.000000	NaN	NaN	0.420000	0.000000	0.000
25%	223.500000	0.000000	2.000000	NaN	NaN	20.125000	0.000000	0.000
50%	446.000000	0.000000	3.000000	NaN	NaN	28.000000	0.000000	0.000
75%	668.500000	1.000000	3.000000	NaN	NaN	38.000000	1.000000	0.000
max	891.000000	1.000000	3.000000	NaN	NaN	80.000000	8.000000	6.000

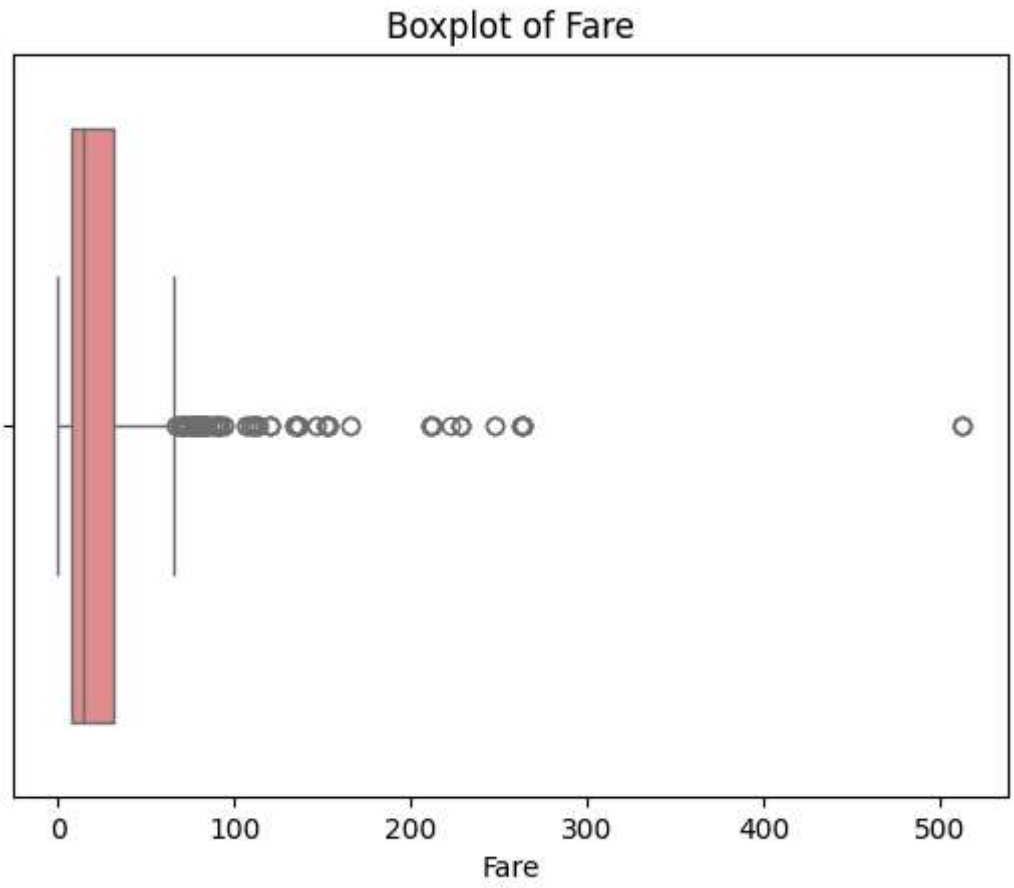
Observation: Missing data exists in Age, Cabin, and Embarked. Survived is the target variable (0 = No, 1 = Yes). Categorical: Sex, Embarked, Pclass; Numerical: Age, Fare, SibSp, Parch.

In [10]:
```
num_cols = ['Age', 'Fare']
train[num_cols].hist(bins=20, figsize=(10, 5), color='skyblue')
plt.suptitle('Histograms of Numerical Columns')
plt.show()
```



In [11]:
```
for col in num_cols:
    sns.boxplot(x=train[col], color='lightcoral')
    plt.title(f'Boxplot of {col}')
    plt.show()
```

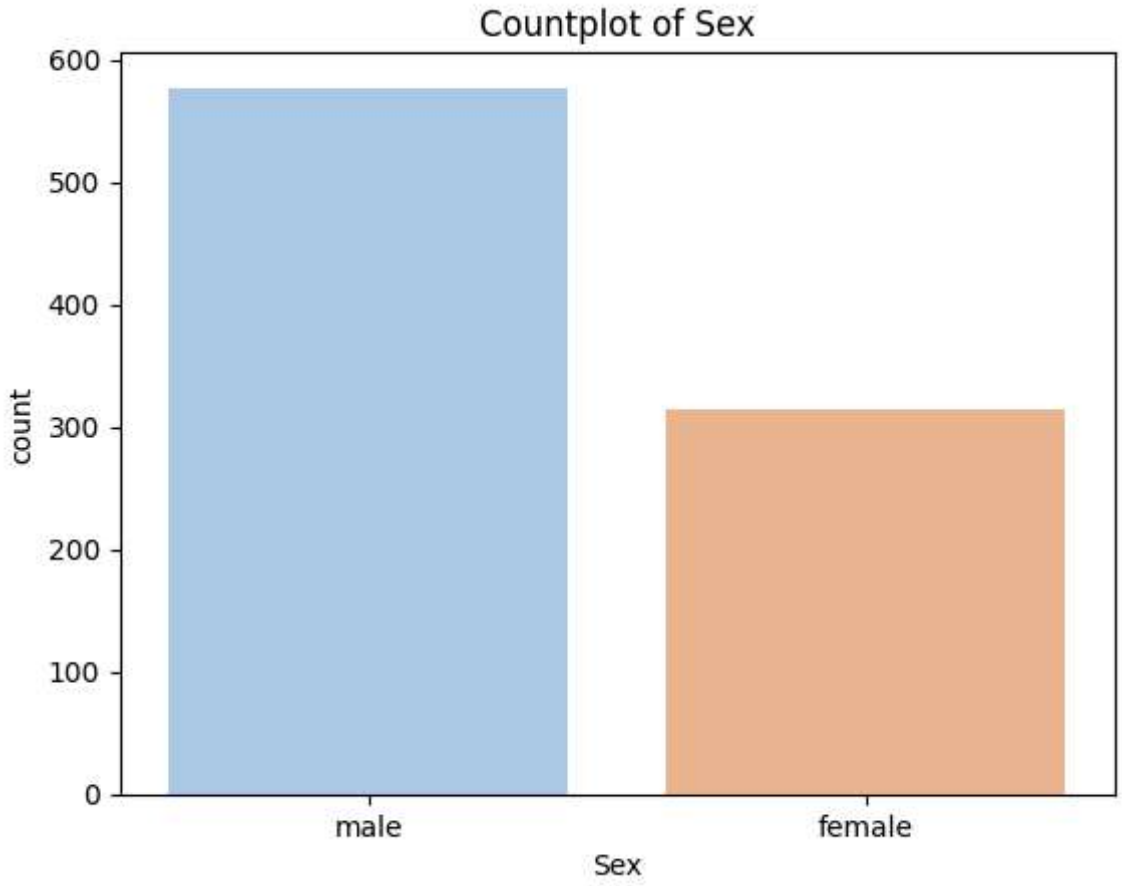




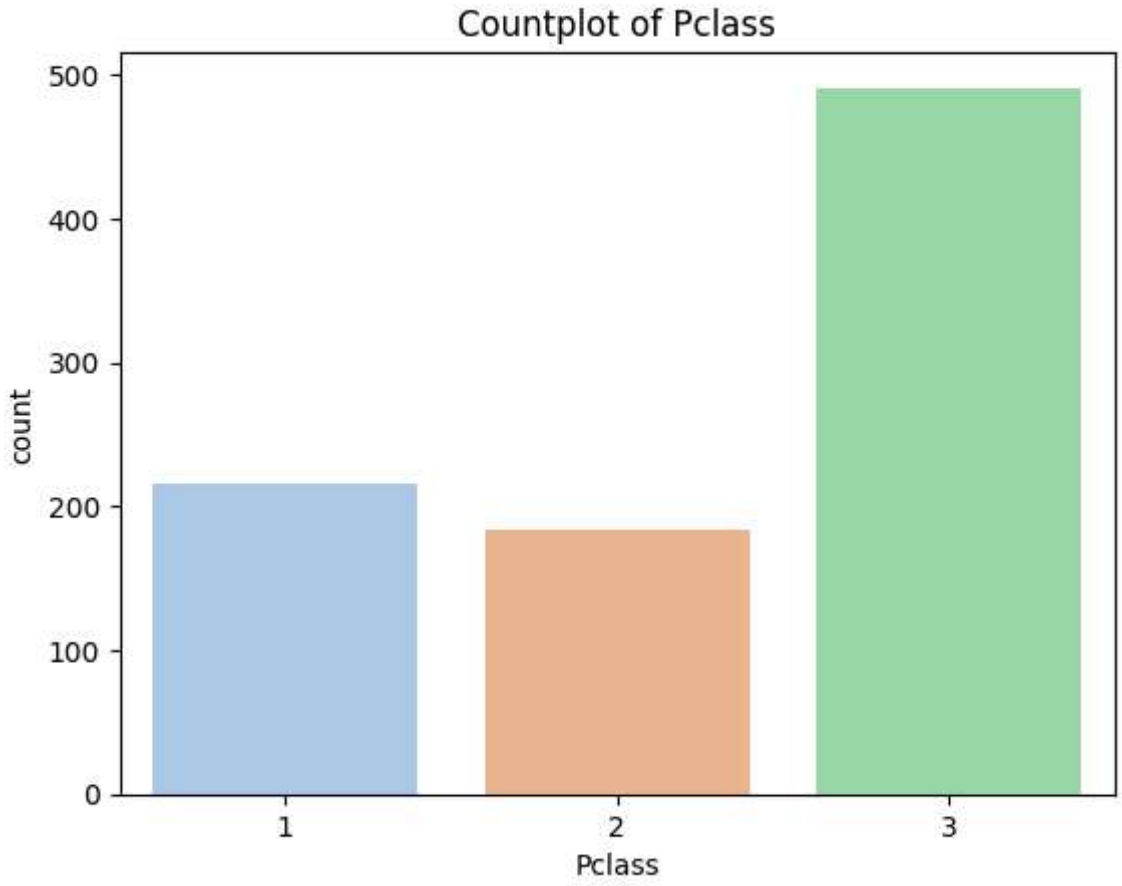
Observation: Fare is heavily right-skewed (a few very high fares). Age is roughly normal with minor outliers.

In [12]:
```
cat_cols = ['Sex', 'Pclass', 'Embarked', 'Survived']

for col in cat_cols:
    sns.countplot(data=train, x=col, palette='pastel')
    plt.title(f'Countplot of {col}')
    plt.show()
```

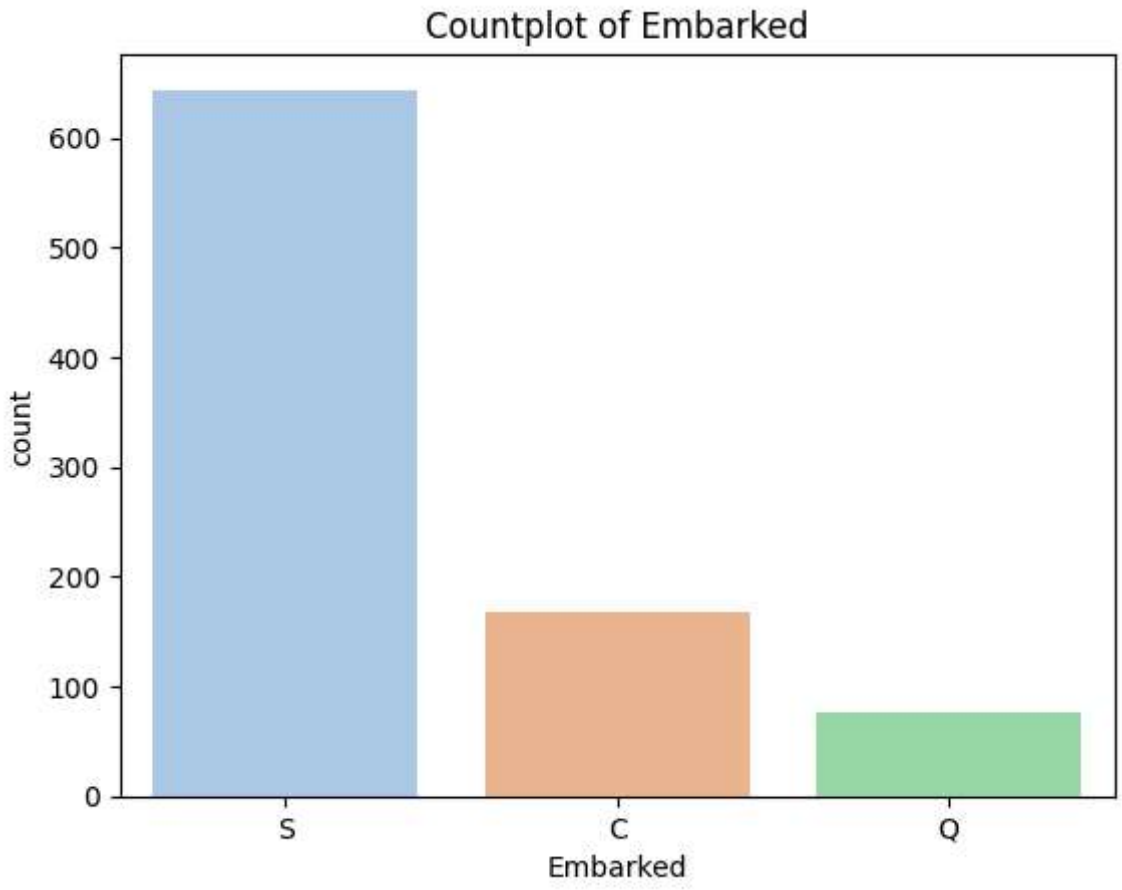
```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\819149248.py:4: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(data=train, x=col, palette='pastel')
```



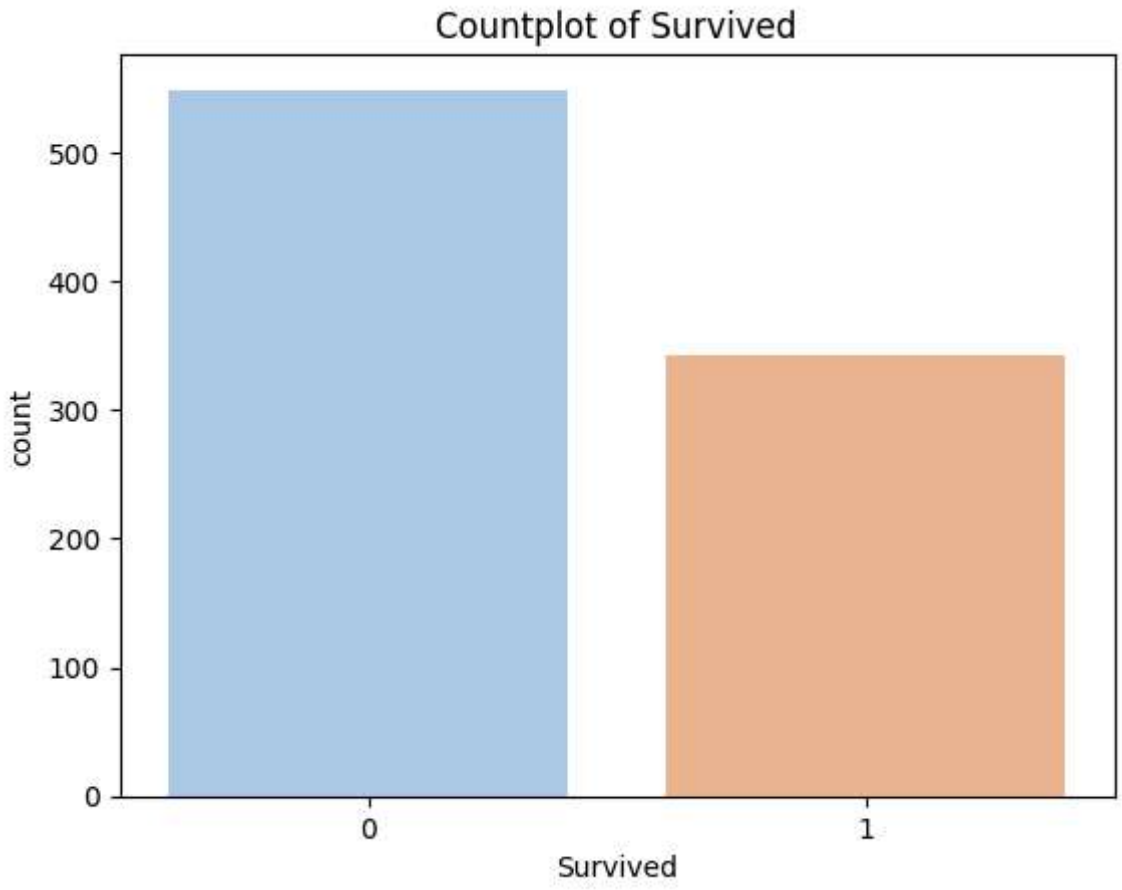

```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\819149248.py:4: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(data=train, x=col, palette='pastel')
```




```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\819149248.py:4: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(data=train, x=col, palette='pastel')
```



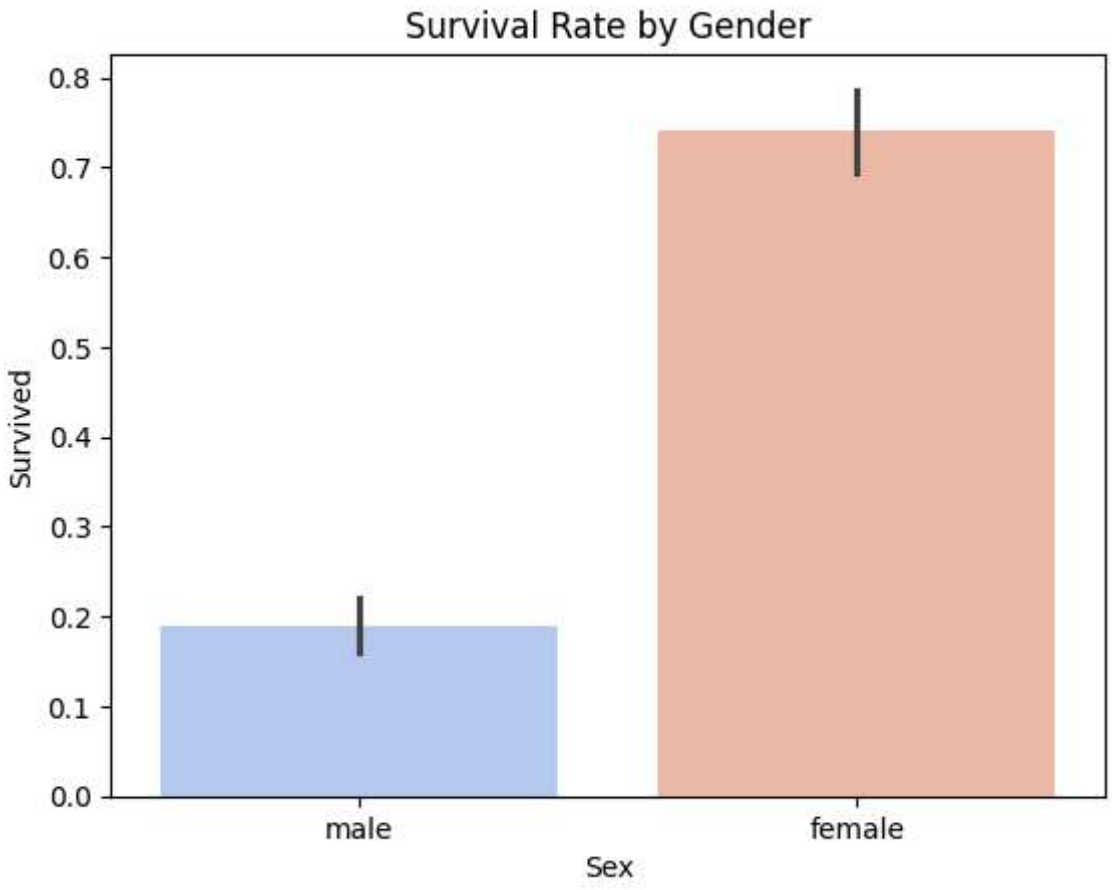

```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\819149248.py:4: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(data=train, x=col, palette='pastel')
```



Observation: More males than females on board. Most passengers in 3rd class. Fewer people embarked at 'Q'.

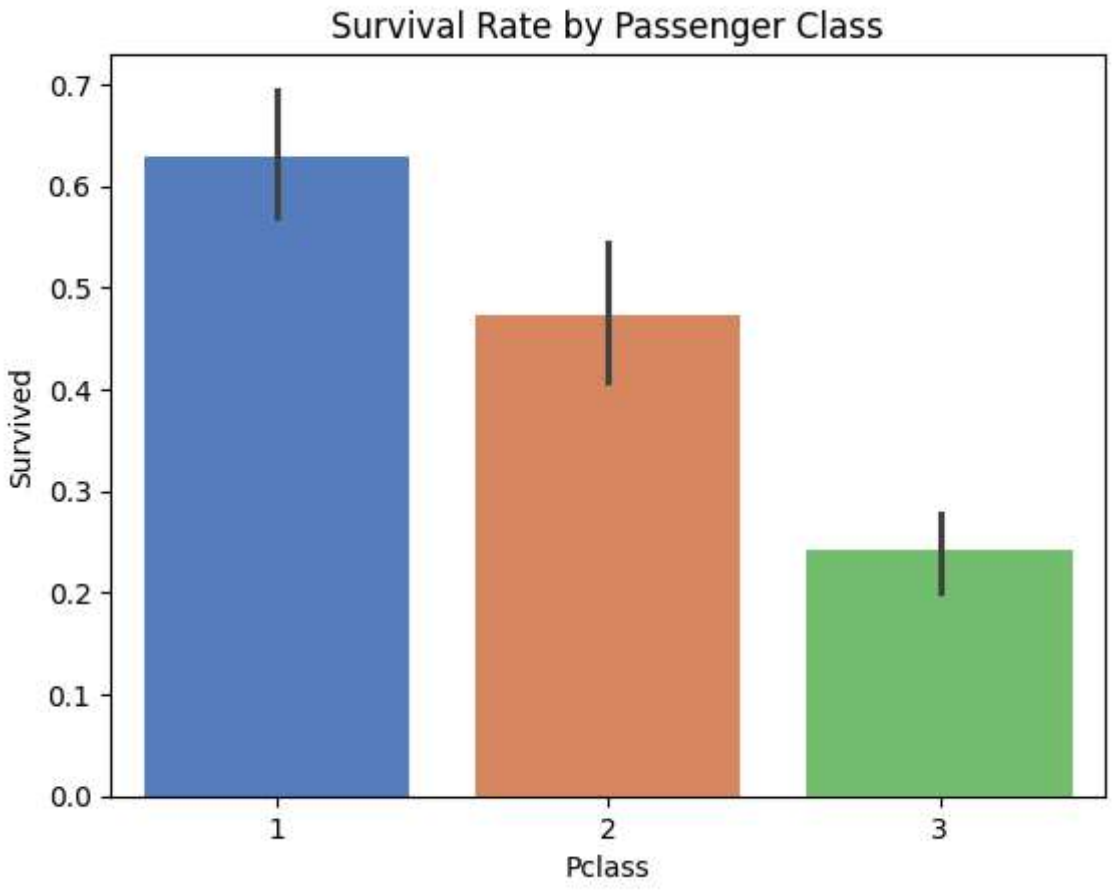
In [13]:
```
sns.barplot(x='Sex', y='Survived', data=train, palette='coolwarm')
plt.title('Survival Rate by Gender')
plt.show()
```

```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\568103199.py:1: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x='Sex', y='Survived', data=train, palette='coolwarm')
```

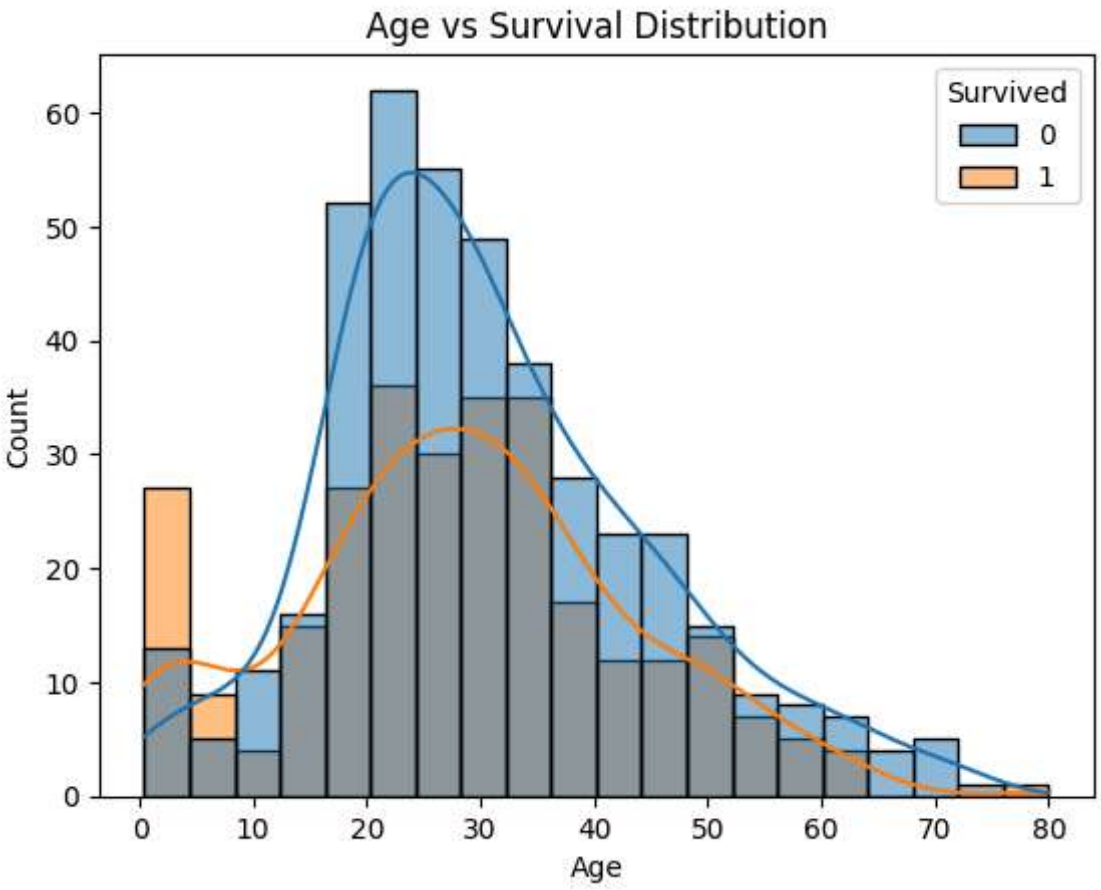


In [14]:
```
sns.barplot(x='Pclass', y='Survived', data=train, palette='muted')
plt.title('Survival Rate by Passenger Class')
plt.show()
```

```
C:\Users\TEMP.SAKSHICM\AppData\Local\Temp\ipykernel_11008\597455925.py:1: FutureWarn
ing:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x='Pclass', y='Survived', data=train, palette='muted')
```

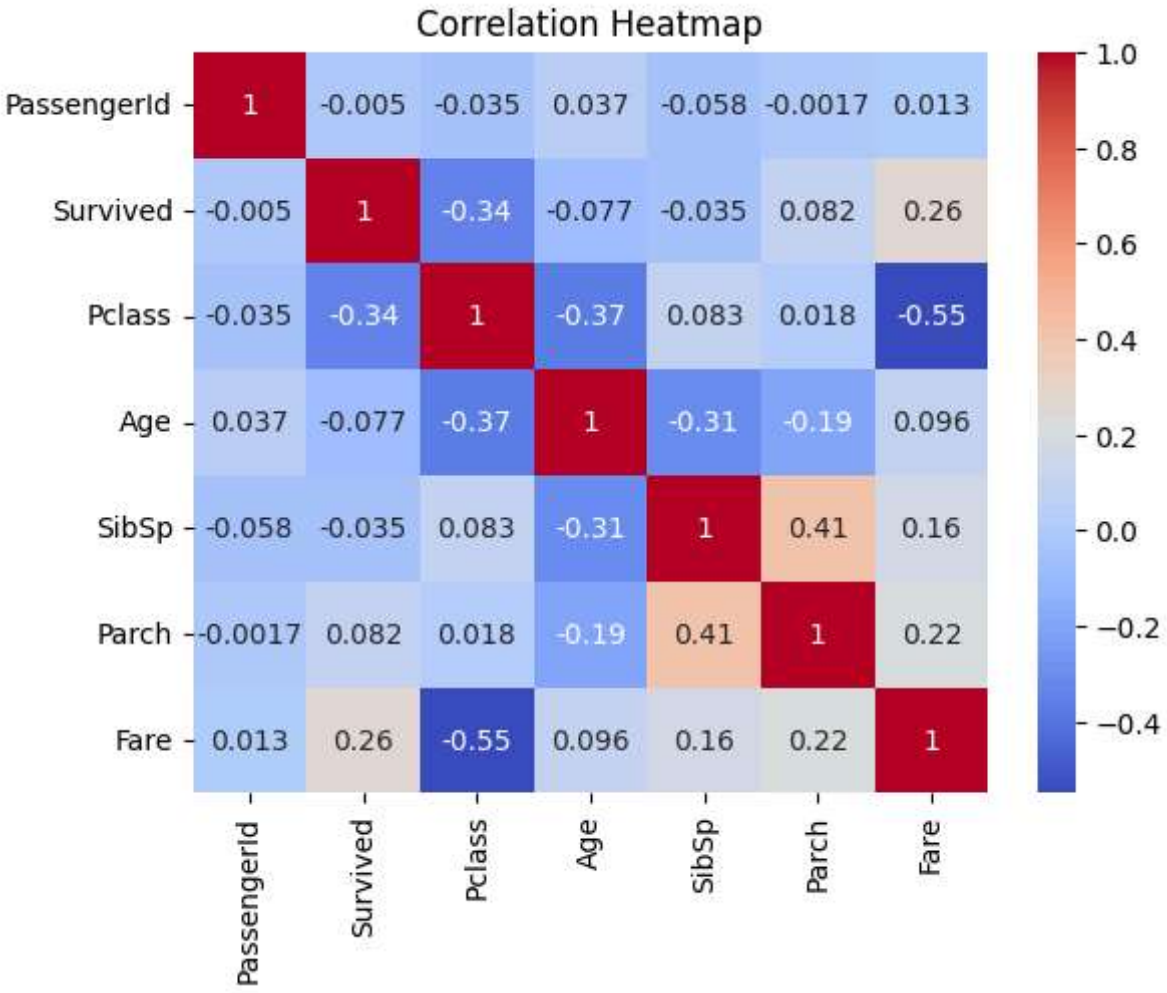


In [15]:
```
sns.histplot(data=train, x='Age', hue='Survived', bins=20, kde=True)
plt.title('Age vs Survival Distribution')
plt.show()
```

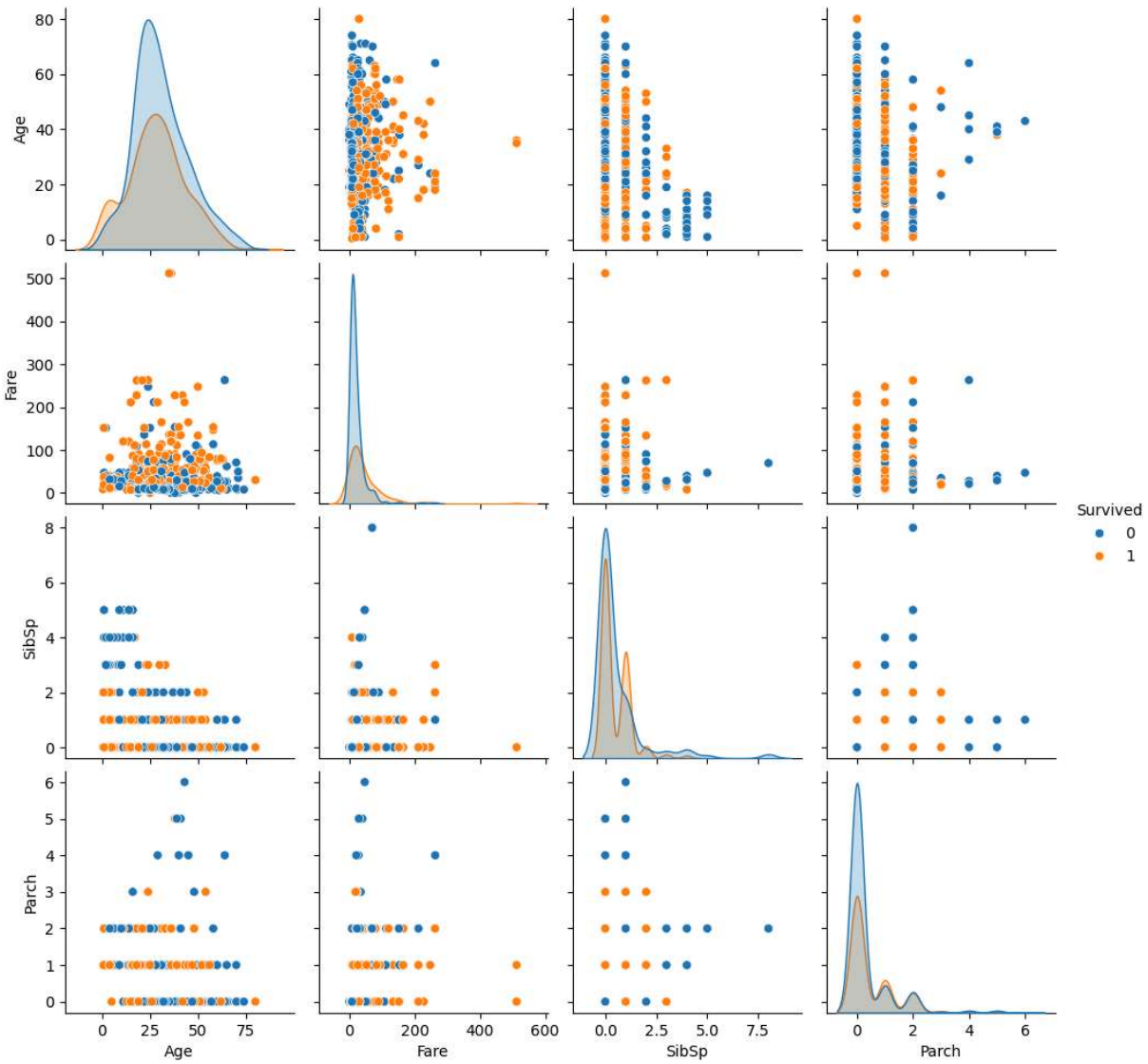


Observation: Females had higher survival rates. 1st class passengers survived more often. Many young children survived (lower age group).

In [16]:
```
corr = train.corr(numeric_only=True)
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```



In [17]:
```
sns.pairplot(train[['Age','Fare','SibSp','Parch','Survived']], hue='Survived')
plt.show()
```



Observation: Pclass and Fare are inversely correlated. Fare has positive correlation with survival. No strong multicollinearity among numeric columns.

In [18]:
```
from IPython.display import Markdown as md

md("""
### 📊 Summary of Titanic EDA Findings

1. **Missing Values:** Found in `Age`, `Embarked`, and `Cabin`.
2. **Gender:** Females had a much higher survival rate than males.
3. **Class:** 1st class passengers survived more than 2nd and 3rd.
4. **Age:** Younger passengers had slightly better survival chances.
5. **Fare:** High fare often associated with higher survival (upper class).
6. **Correlation:** No strong multicollinearity; Fare shows moderate correlation wi
7. **Skewness:** Fare is right-skewed but can be normalized using log transformatio
""")
```

Out[18]:

📊 Summary of Titanic EDA Findings

1. **Missing Values:** Found in `Age`, `Embarked`, and `Cabin`.
2. **Gender:** Females had a much higher survival rate than males.
3. **Class:** 1st class passengers survived more than 2nd and 3rd.
4. **Age:** Younger passengers had slightly better survival chances.
5. **Fare:** High fare often associated with higher survival (upper class).
6. **Correlation:** No strong multicollinearity; Fare shows moderate correlation with Survived.
7. **Skewness:** Fare is right-skewed but can be normalized using log transformation.

In [ ]:

In [ ]: