Task 2: Data Cleaning and Exploratory Data Analysis (EDA)

In this task, we perform data cleaning and exploratory data analysis on the Titanic dataset.
Our goal is to understand data structure, clean missing values, and explore relationships between variables to discover patterns and trends.

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

In [2]:
```
url = 'https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic
df = pd.read_csv(url)
df.head()
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	F
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0

In [3]:
```
df.isnull().sum()
```

Out[3]:
```
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 [6]:
```
print(df.columns)
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Embarked'],
      dtype='object')
```

In [7]:
```
if 'Cabin' in df.columns:
    df = df.drop(columns=['Cabin'])

df = df.dropna(subset=['Age', 'Embarked'])
```

In [8]:
```
df.info()
df.describe()
```

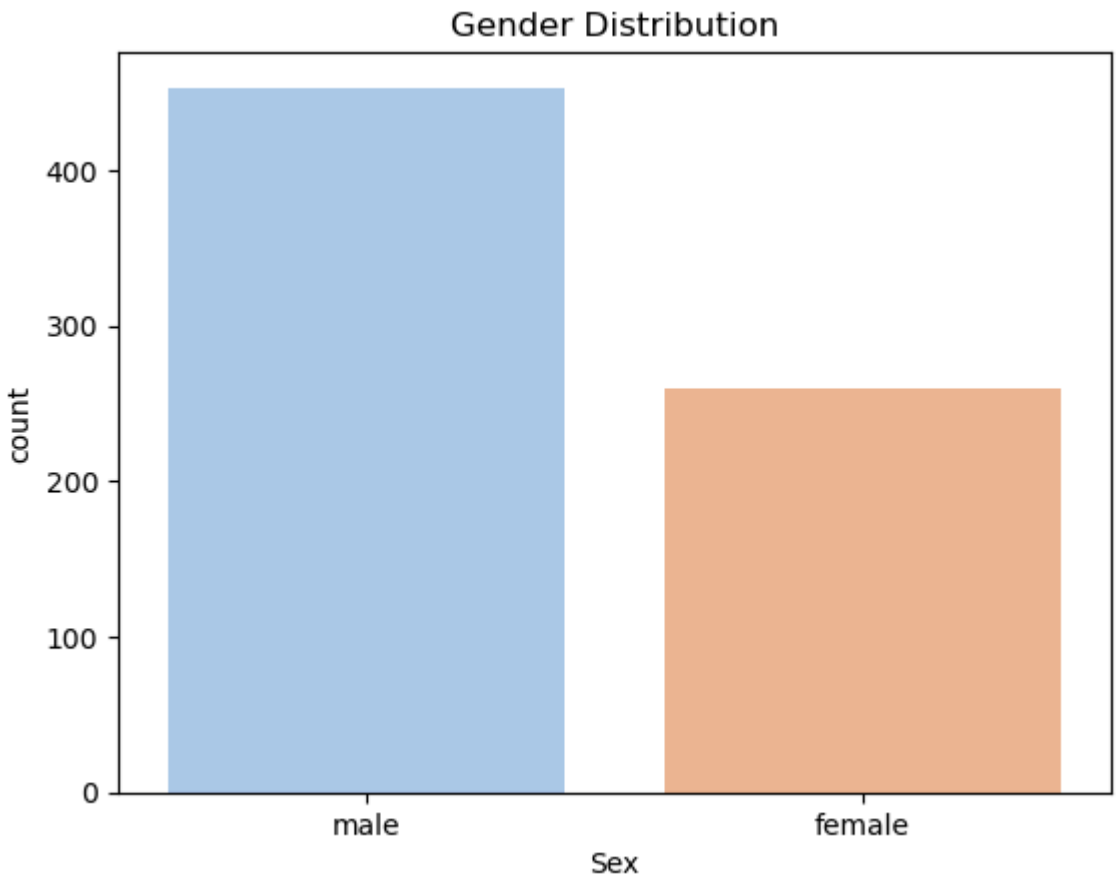
```
<class 'pandas.core.frame.DataFrame'>
Index: 712 entries, 0 to 890
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  712 non-null    int64
 1   Survived     712 non-null    int64
 2   Pclass       712 non-null    int64
 3   Name         712 non-null    object
 4   Sex          712 non-null    object
 5   Age          712 non-null    float64
 6   SibSp        712 non-null    int64
 7   Parch        712 non-null    int64
 8   Ticket       712 non-null    object
 9   Fare         712 non-null    float64
 10  Embarked     712 non-null    object
dtypes: float64(2), int64(5), object(4)
memory usage: 66.8+ KB
```

Out[8]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	
count	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000	712.00
mean	448.589888	0.404494	2.240169	29.642093	0.514045	0.432584	34.567
std	258.683191	0.491139	0.836854	14.492933	0.930692	0.854181	52.938
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.00
25%	222.750000	0.000000	1.000000	20.000000	0.000000	0.000000	8.05
50%	445.000000	0.000000	2.000000	28.000000	0.000000	0.000000	15.645
75%	677.250000	1.000000	3.000000	38.000000	1.000000	1.000000	33.00
max	891.000000	1.000000	3.000000	80.000000	5.000000	6.000000	512.329

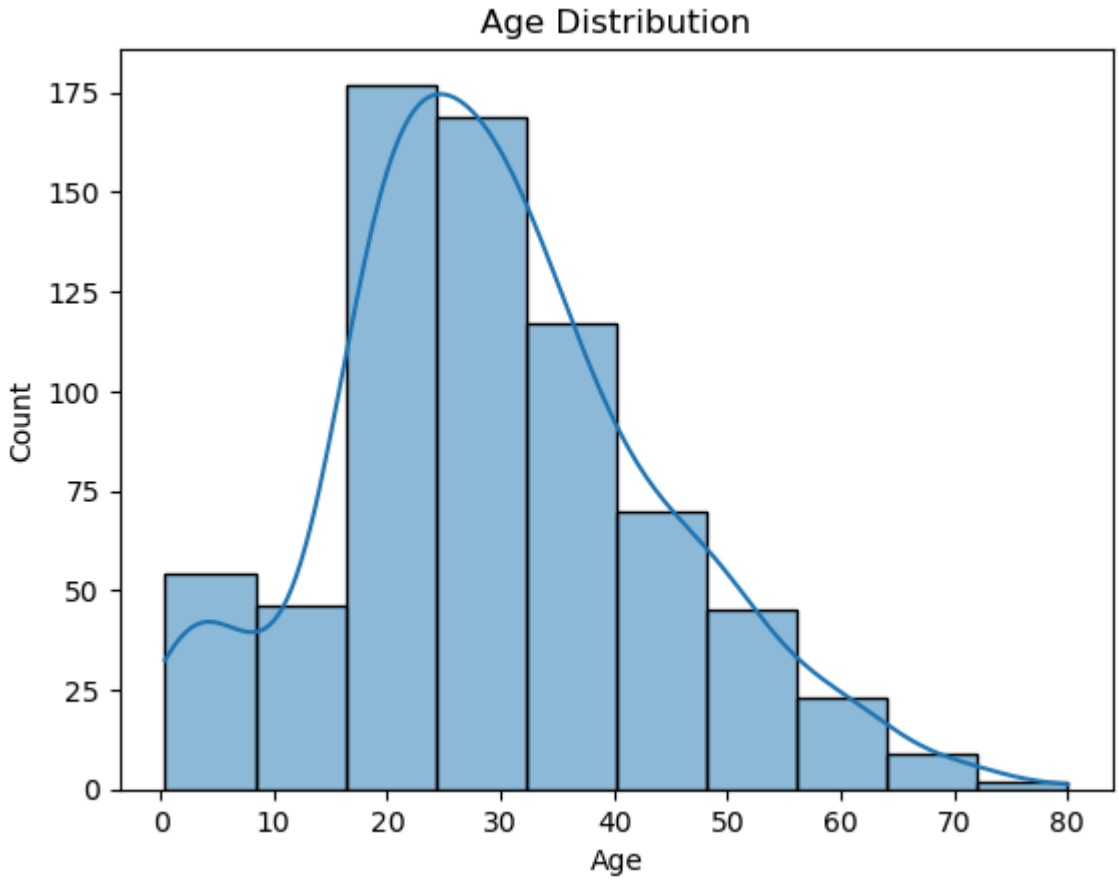
In [9]:

```
sns.countplot(data=df, x='Sex', palette='pastel')
plt.title('Gender Distribution')
plt.show()
```

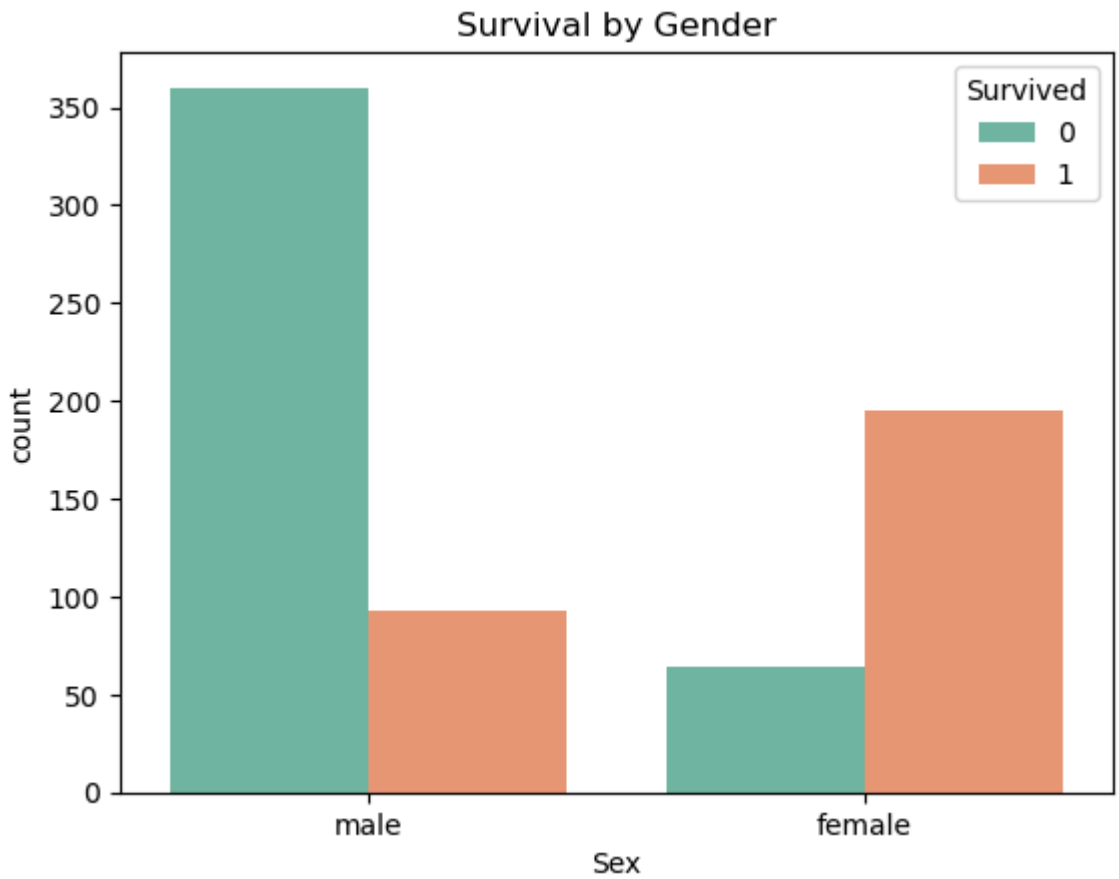


In [10]:

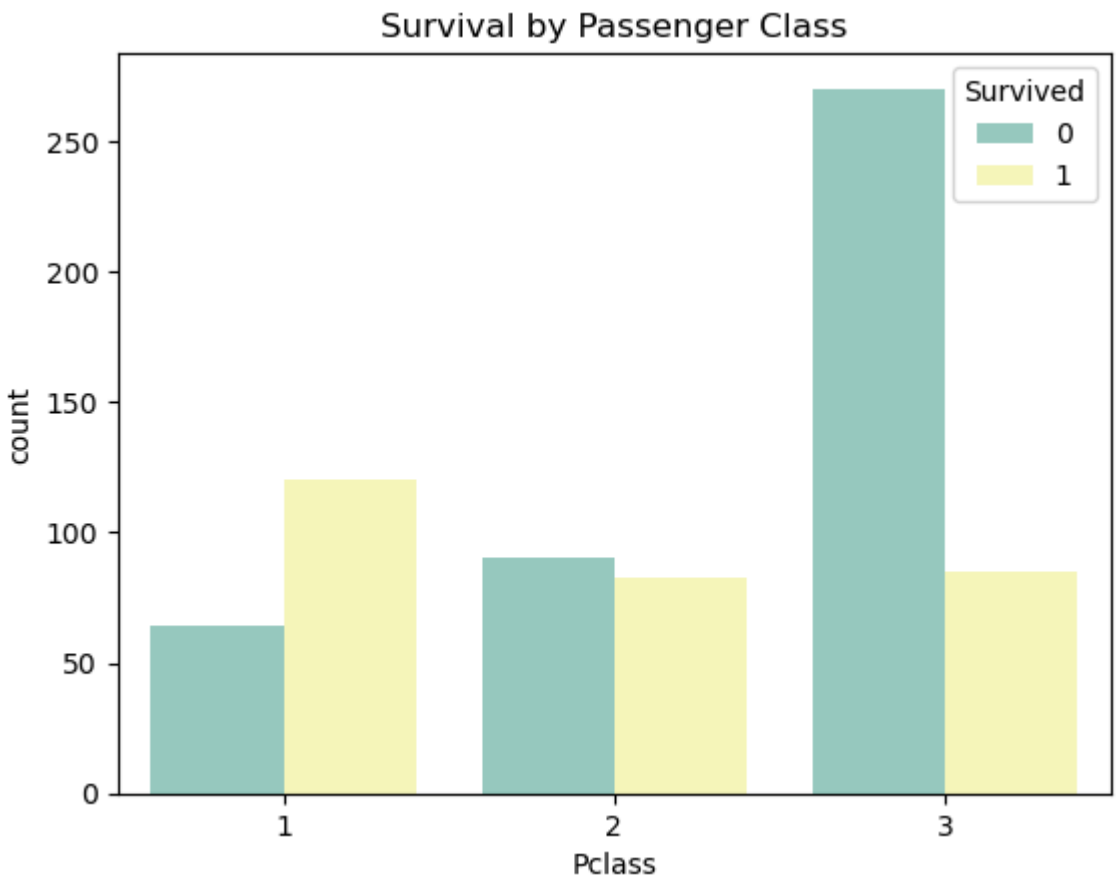
```
sns.histplot(df['Age'], bins=10, kde=True)
plt.title('Age Distribution')
plt.show()
```



In [11]:
```
sns.countplot(data=df, x='Sex', hue='Survived', palette='Set2')
plt.title('Survival by Gender')
plt.show()
```



In [12]:
```
sns.countplot(data=df, x='Pclass', hue='Survived', palette='Set3')
plt.title('Survival by Passenger Class')
plt.show()
```



Conclusion

- The dataset had some missing values, especially in the "Cabin" and "Age" columns.
- Most passengers were male, and most were in third class.
- Females had a higher survival rate than males.
- Passengers in first class had a better survival rate than those in second and third.

In [ ]: