```
In [4]:  import pandas as pd
         import numpy as np
         import seaborn as sns
         import matplotlib.pyplot as plt
```

```
In [5]:  df = pd.read_csv('train.csv')   # Replace with your actual dataset path if differ
```

```
In [6]:  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]:  df.describe()
```

Out[7]:

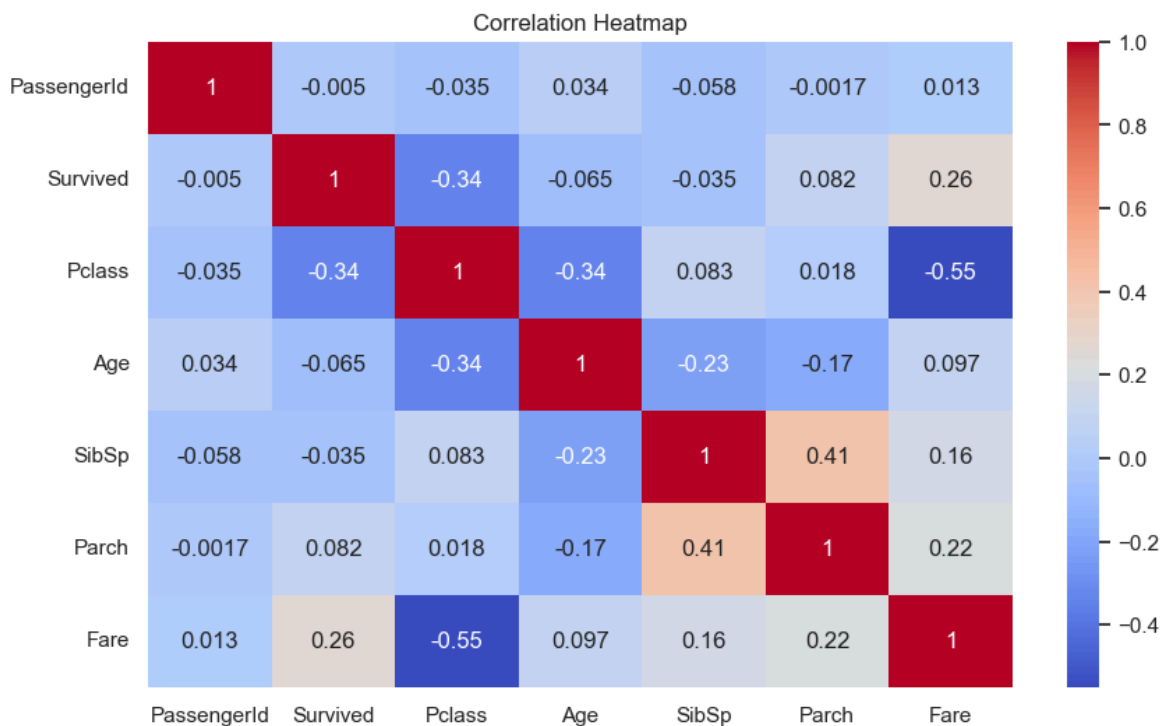|	PassengerId	Survived	Pclass	Age	SibSp	Parch	
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.00(
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.00(
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.91(
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.00(
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329

```
In [11]:  df['Survived'].value_counts()
          df['Pclass'].value_counts()
          df['Sex'].value_counts()
```

```
Out[11]:  Sex
          male      577
          female    314
          Name: count, dtype: int64
```

```
In [12]: #Pairplot: Survivors are more frequent in higher classes and younger ages.
         sns.pairplot(df[['Survived','Pclass','Age','Fare']])
         plt.show()
```
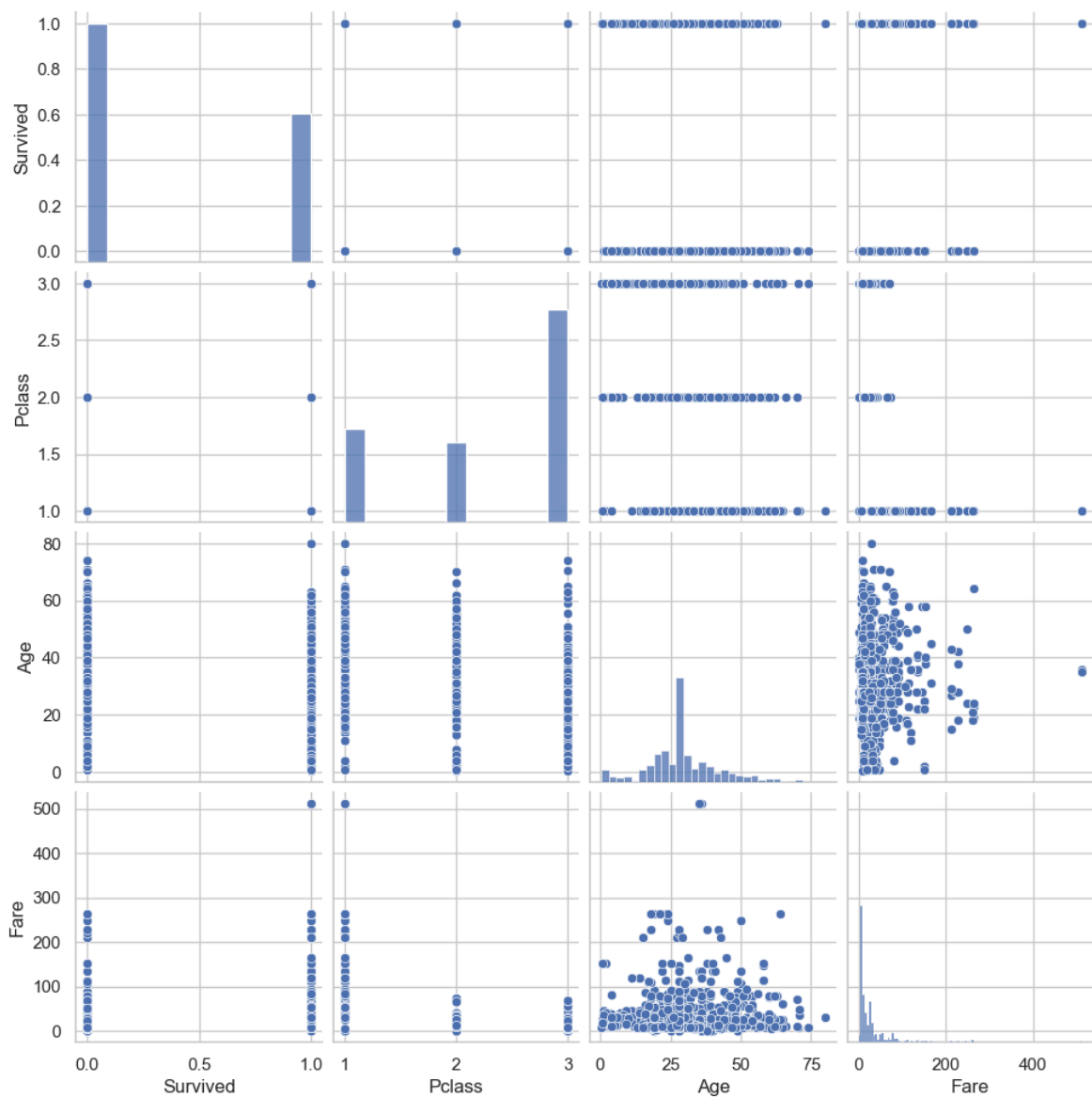


```
In [13]: corr = df.corr(numeric_only=True)
         sns.heatmap(corr, annot=True, cmap='coolwarm')
         plt.title('Correlation Heatmap')
         plt.show()
```

Correlation Heatmap

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1	-0.005	-0.035	0.034	-0.058	-0.0017	0.013
Survived	-0.005	1	-0.34	-0.065	-0.035	0.082	0.26
Pclass	-0.035	-0.34	1	-0.34	0.083	0.018	-0.55
Age	0.034	-0.065	-0.34	1	-0.23	-0.17	0.097
SibSp	-0.058	-0.035	0.083	-0.23	1	0.41	0.16
Parch	-0.0017	0.082	0.018	-0.17	0.41	1	0.22
Fare	0.013	0.26	-0.55	0.097	0.16	0.22	1

```
In [14]: df['Age'].hist(bins=30, color='skyblue', edgecolor='black')
         plt.title('Age Distribution')
         plt.xlabel('Age')
         plt.ylabel('Frequency')
         plt.grid(False)
         plt.show()
```

Age Distribution
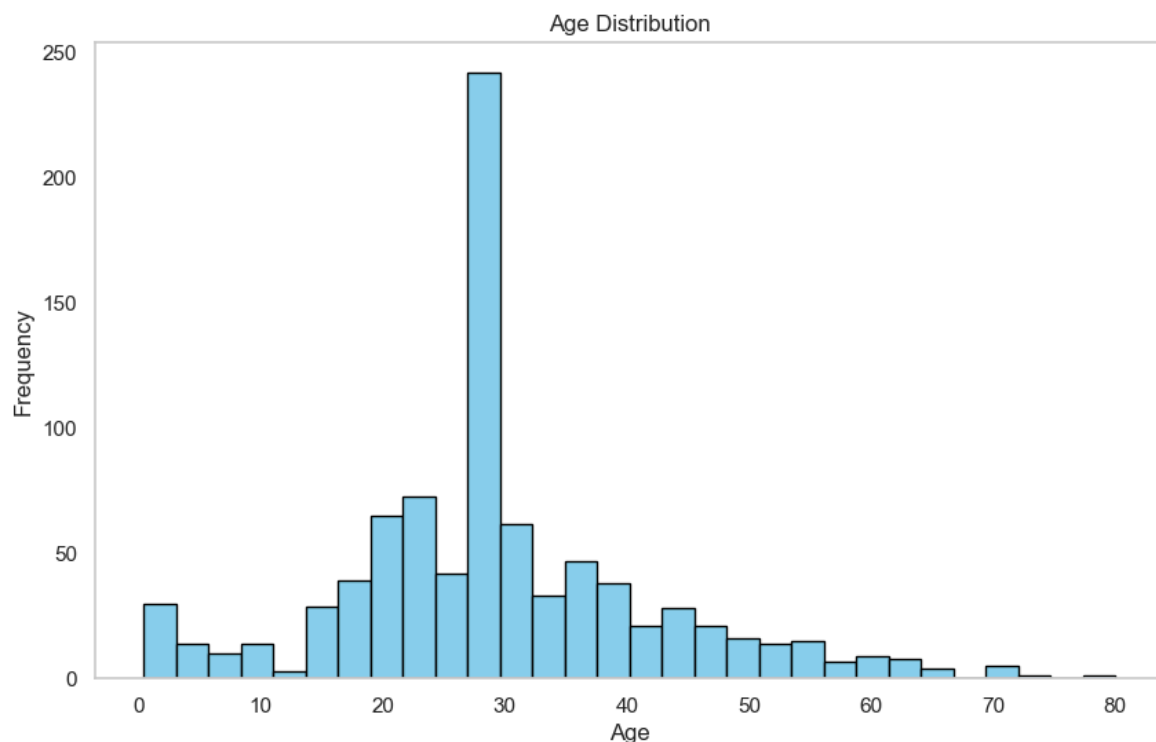
```
In [16]: sns.boxplot(x='Pclass', y='Age', data=df, hue='Pclass', palette='pastel', legend
         plt.title('Age by Passenger Class')
         plt.xlabel('Passenger Class')
         plt.ylabel('Age')
         plt.show()
```

Age by Passenger Class
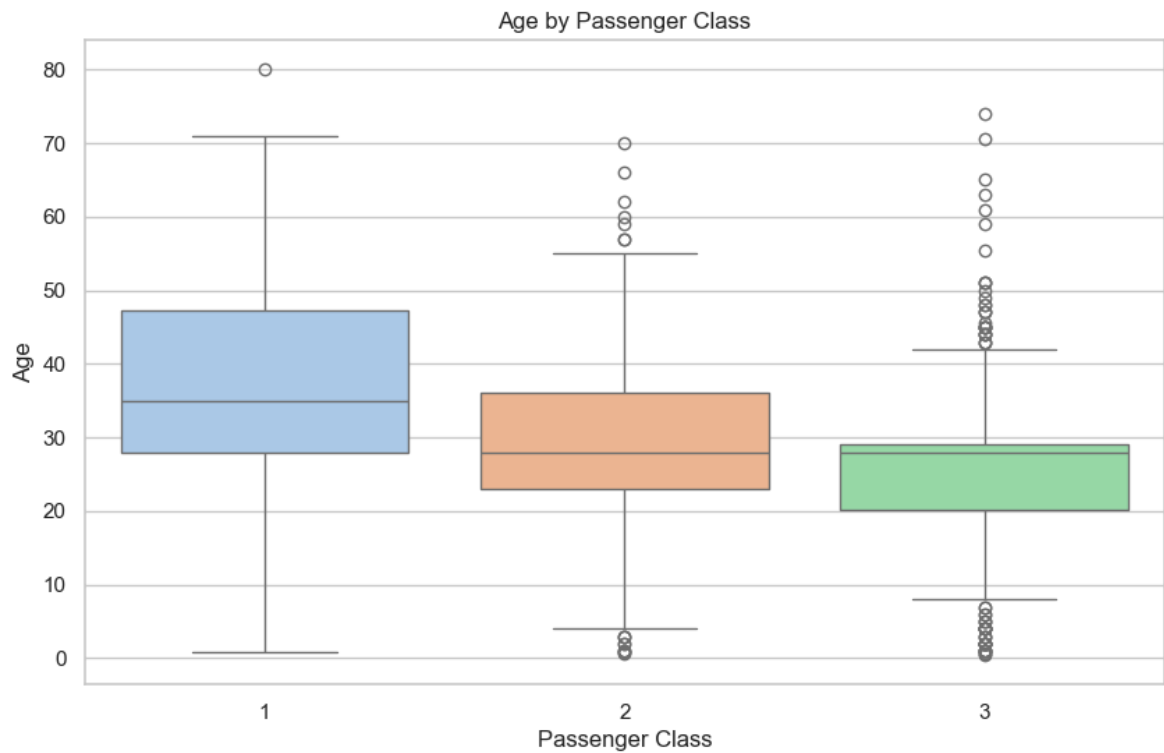

```
In [17]:  plt.figure(figsize=(8, 5))
          plt.scatter(df['Age'], df['Fare'], alpha=0.6, edgecolors='w', color='teal')
          plt.title('Fare vs Age')
          plt.xlabel('Age')
          plt.ylabel('Fare')
          plt.grid(True)
          plt.show()
```

Fare vs Age