```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

sns.set(style="whitegrid") # for better visuals
```

```
titanic = pd.read_csv('train.csv')
titanic.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fa
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.28

Next steps: Generate code with titanic | View recommended plots | New interactive sheet

```
print(titanic.shape)
print(titanic.info())
print(titanic.isnull().sum())
```

```
(891, 12)
<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
```

```
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
```

```
titanic.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fa
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000(
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	0.000(
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910-
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	31.000(
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329:

Observation:

- The average passenger age is around 29.7 years.
- Fare values are highly skewed with some very high ticket prices.
- 50% of passengers paid fare amounts less than 14.45.

```
titanic['Sex'].value_counts()
titanic['Embarked'].value_counts()
```

Embarked	count
S	644
C	168
Q	77

dtype: int64

Observation:

- There are more males (577) than females (314) aboard.
- Most passengers embarked from Southampton ('S'), followed by Cherbourg ('C') and Queenstown ('Q').

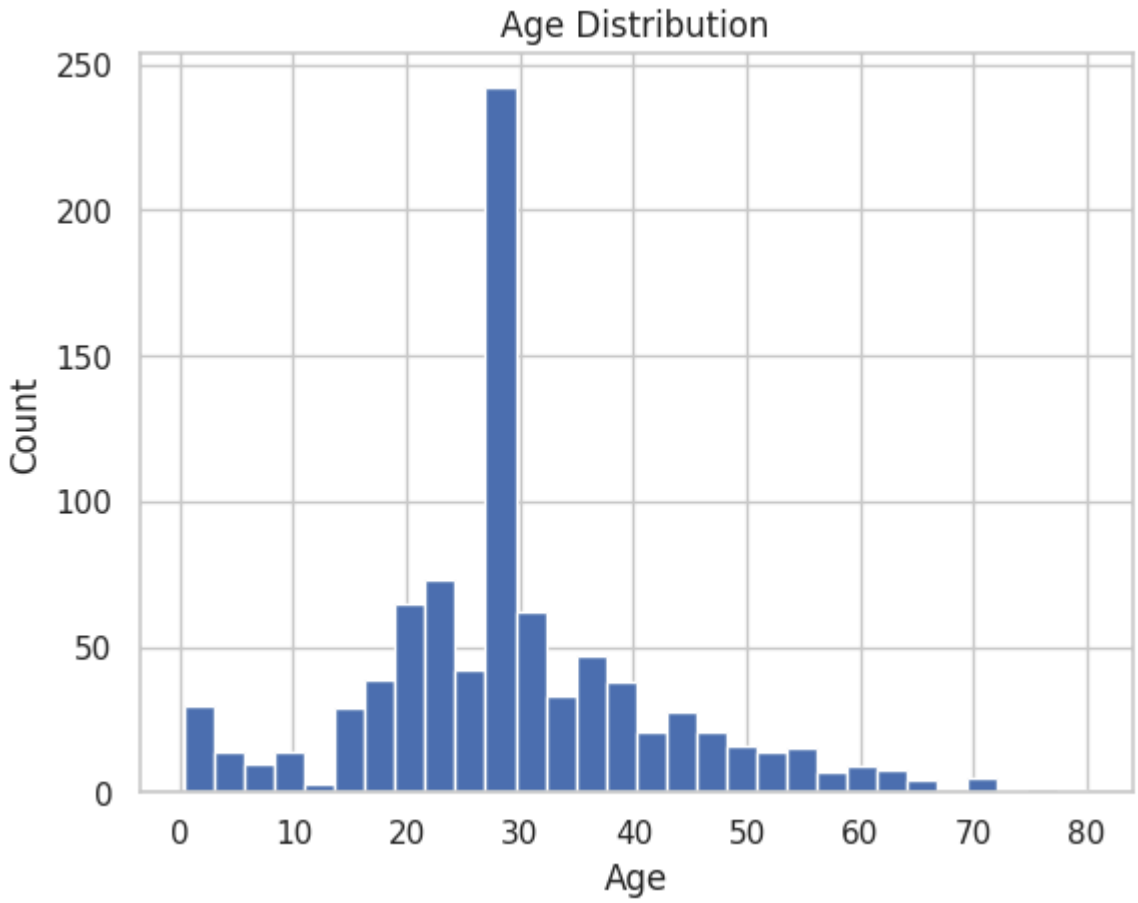
```
titanic['Age'].fillna(titanic['Age'].median(), inplace=True)
titanic['Embarked'].fillna(titanic['Embarked'].mode()[0], inplace=True)
```

```
ipython-input-12-595430d40845>:1: FutureWarning: A value is trying to be set on a copy
he behavior will change in pandas 3.0. This inplace method will never work because the

or example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({co

 titanic['Age'].fillna(titanic['Age'].median(), inplace=True)
```

Observation:

- Missing 'Age' values were filled with the median age (28.0).
- Missing 'Embarked' values were filled with the mode, which is 'S' (Southampton).
- 'Cabin' still has many missing values and may need further treatment if required.

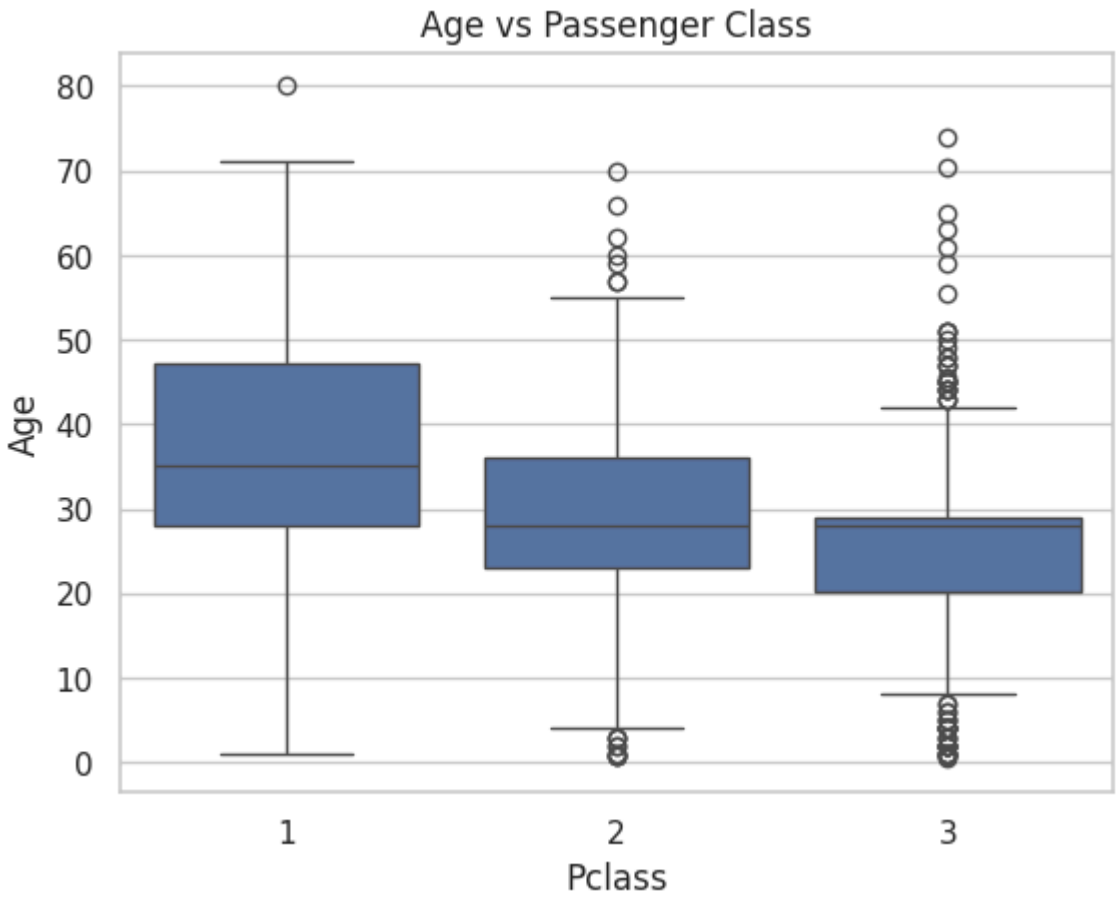
```
titanic['Age'].hist(bins=30)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
```



Observation:

- Most passengers were between 20 to 40 years old.
- Very few very young or very old passengers were on board.

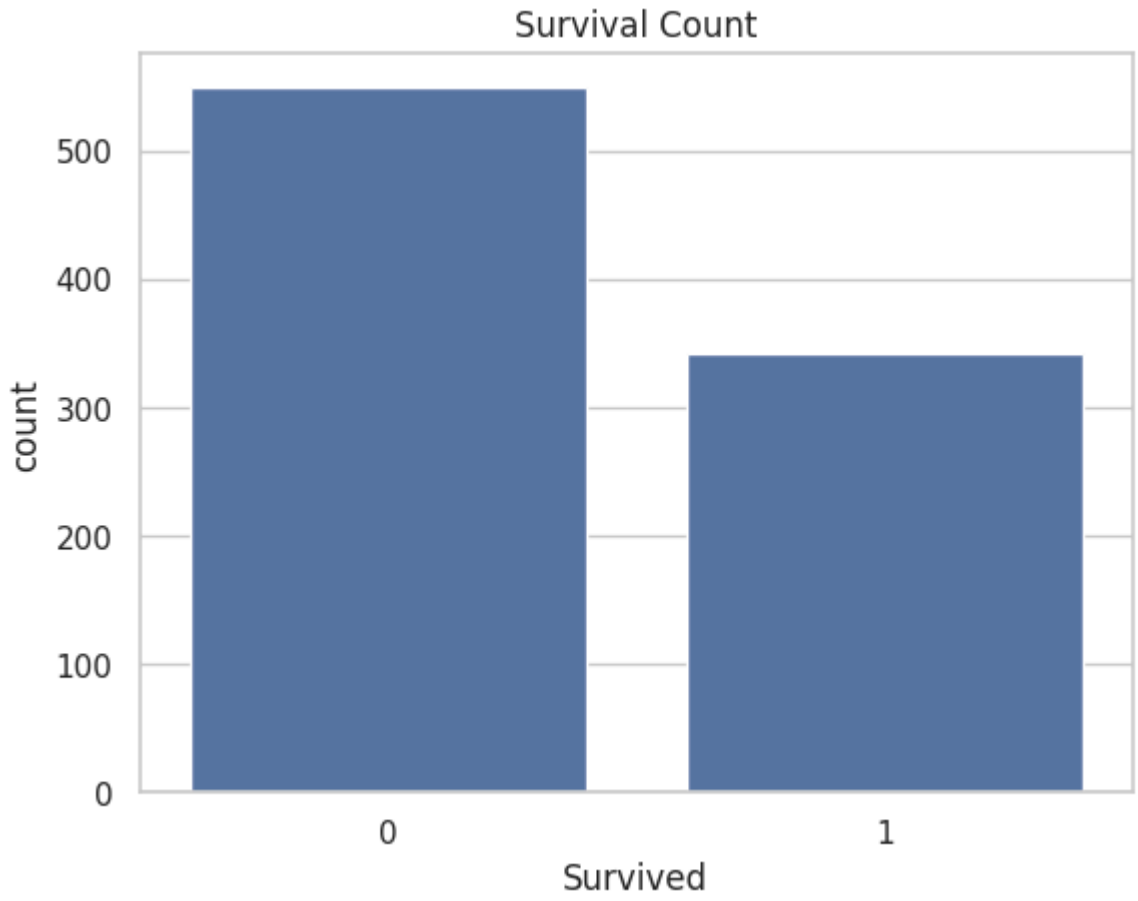
```
sns.boxplot(x='Pclass', y='Age', data=titanic)
plt.title('Age vs Passenger Class')
plt.show()
```



Observation:

- 1st class passengers are generally older than 2nd and 3rd class passengers.
- 3rd class had a wider range of younger passengers.

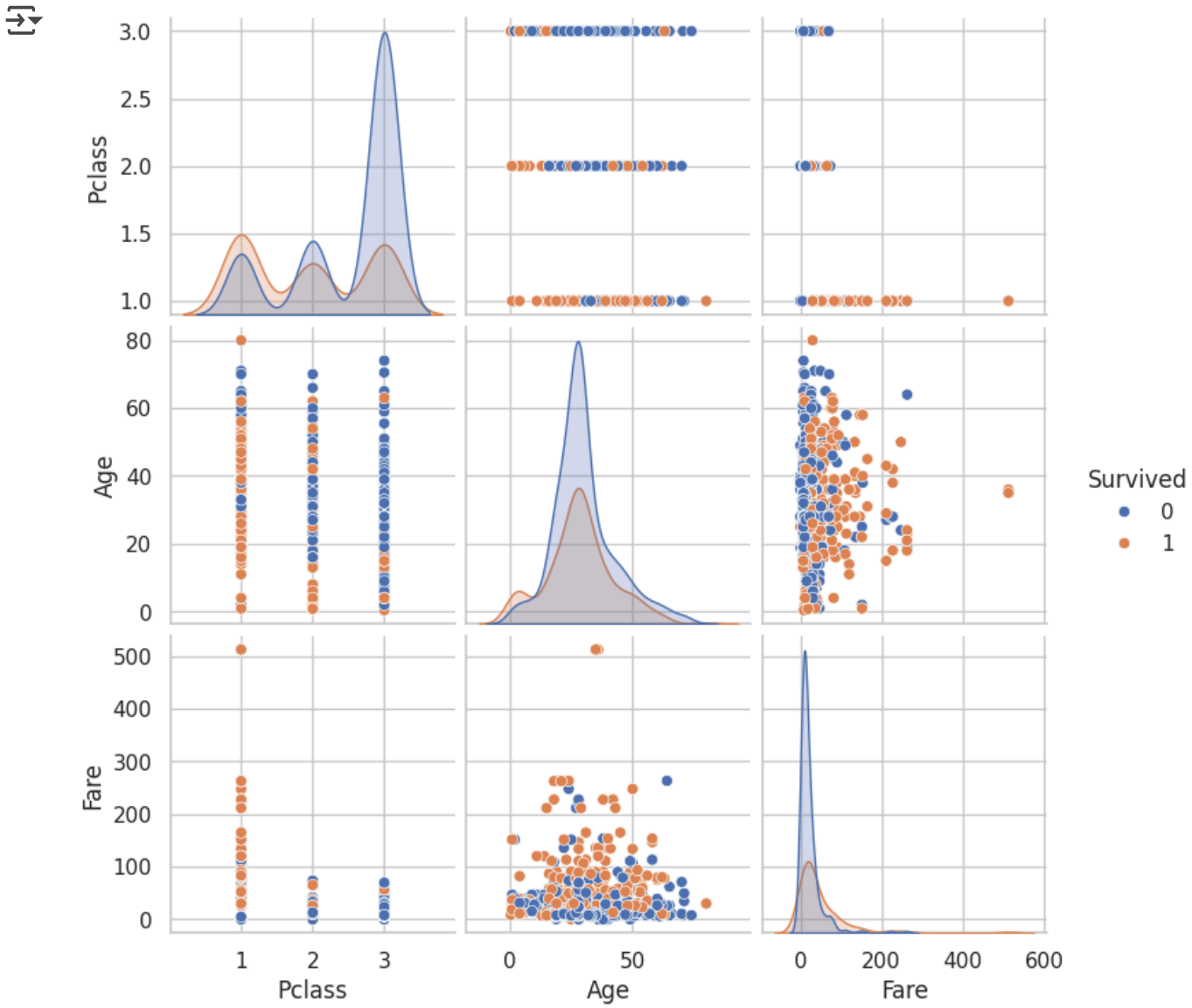
```
sns.countplot(x='Survived', data=titanic)
plt.title('Survival Count')
plt.show()
```



Observation:

- More passengers died (Survived = 0) than survived (Survived = 1).
- Survival rate was less than 50%.

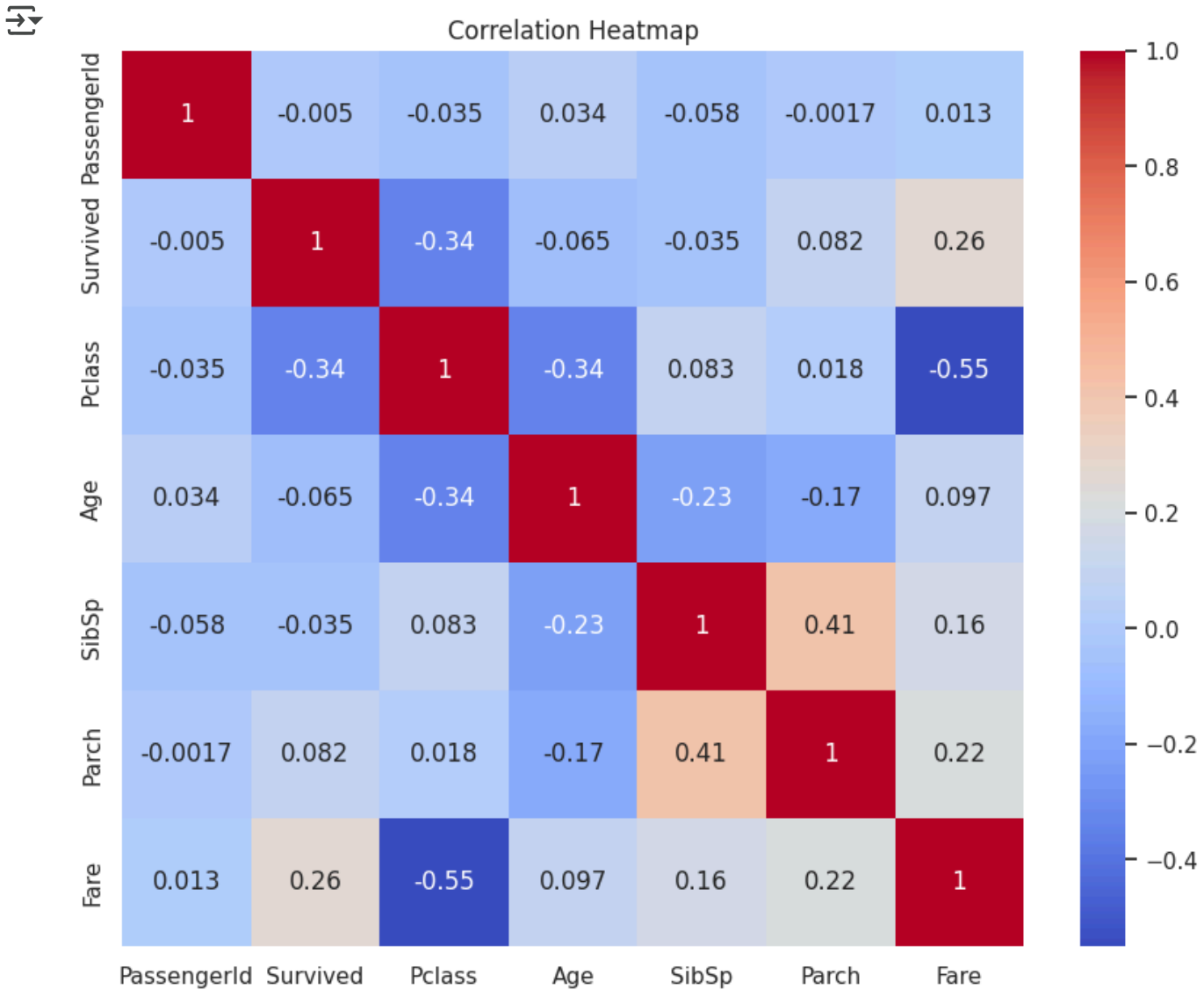
```
sns.pairplot(titanic[['Survived', 'Pclass', 'Age', 'Fare']], hue='Survived')
plt.show()
```



Observation:

- Higher fares and 1st class are strongly associated with higher survival rates.
- Passengers who survived were generally younger and had paid higher fares.

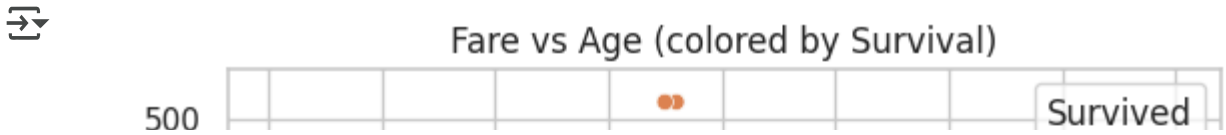
```
plt.figure(figsize=(10,8))
# Select only numerical features for correlation analysis
numerical_features = titanic.select_dtypes(include=np.number)
sns.heatmap(numerical_features.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```



Observation:

- "Fare" and "Pclass" are negatively correlated (higher class = lower Pclass number = higher fare).
- "Survived" shows a good positive correlation with "Fare" and slight negative correlation with "Pclass".

```
sns.scatterplot(x='Age', y='Fare', hue='Survived', data=titanic)
plt.title('Fare vs Age (colored by Survival)')
plt.show()
```


Fare vs Age (colored by Survival)
500
Survived