In [1]:
```python
from mpl_toolkits.mplot3d import Axes3D
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt # plotting
import numpy as np # linear algebra
import os # accessing directory structure
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
```

In [4]:
```python
path = "tested.csv"

df = pd.read_csv(path)

df.head(5)
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S

In [5]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  418 non-null    int64
 1   Survived     418 non-null    int64
 2   Pclass       418 non-null    int64
 3   Name         418 non-null    object
 4   Sex          418 non-null    object
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64
 7   Parch        418 non-null    int64
 8   Ticket       418 non-null    object
 9   Fare         417 non-null    float64
 10  Cabin        91 non-null     object
 11  Embarked     418 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 39.3+ KB
```

In [6]:
```python
df.describe
```

Out[6]:
```
<bound method NDFrame.describe of      PassengerId  Survived  Pclass  \
0            892         0       3
1            893         1       3
2            894         0       2
3            895         0       3
4            896         1       3
..           ...       ...     ...
413         1305         0       3
414         1306         1       1
415         1307         0       3
416         1308         0       3
417         1309         0       3

                                             Name     Sex   Age  SibSp  Parch  \
0                                Kelly, Mr. James    male  34.5      0      0
1                Wilkes, Mrs. James (Ellen Needs)  female  47.0      1      0
2                       Myles, Mr. Thomas Francis    male  62.0      0      0
3                                Wirz, Mr. Albert    male  27.0      0      0
4    Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female  22.0      1      1
..                                            ...     ...   ...    ...    ...
413                            Spector, Mr. Woolf    male   NaN      0      0
414                  Oliva y Ocana, Dona. Fermina  female  39.0      0      0
415                  Saether, Mr. Simon Sivertsen    male  38.5      0      0
416                           Ware, Mr. Frederick    male   NaN      0      0
417                     Peter, Master. Michael J    male   NaN      1      1

                 Ticket      Fare Cabin Embarked
0                330911    7.8292   NaN        Q
1                363272    7.0000   NaN        S
2                240276    9.6875   NaN        Q
3                315154    8.6625   NaN        S
4               3101298   12.2875   NaN        S
..                  ...       ...   ...      ...
413           A.5. 3236    8.0500   NaN        S
414            PC 17758  108.9000  C105        C
415  SOTON/O.Q. 3101262    7.2500   NaN        S
416              359309    8.0500   NaN        S
417                2668   22.3583   NaN        C

[418 rows x 12 columns]>
```

In [7]:
```python
df.shape
```

Out[7]:
```
(418, 12)
```

In [9]:
```python
df.dtypes
```

Out[9]:
```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

In [10]:
```python
## Check the nulls
df.isna().sum()
```

Out[10]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin          327
Embarked         0
dtype: int64
```

In [11]:
```python
df.dropna(how='any',inplace=True)
print(df)
```

```
     PassengerId  Survived  Pclass  \
12           904         1       1
14           906         1       1
24           916         1       1
26           918         1       1
28           920         0       1
..           ...       ...     ...
404         1296         0       1
405         1297         0       2
407         1299         0       1
411         1303         1       1
414         1306         1       1

                                                  Name     Sex   Age  SibSp  \
12       Snyder, Mrs. John Pillsbury (Nelle Stevenson)  female  23.0      1
14   Chaffee, Mrs. Herbert Fuller (Carrie Constance...  female  47.0      1
24     Ryerson, Mrs. Arthur Larned (Emily Maria Borie)  female  48.0      1
26                          Ostby, Miss. Helene Ragnhild  female  22.0      0
28                              Brady, Mr. John Bertram    male  41.0      0
..                                                 ...     ...   ...    ...
404                         Frauenthal, Mr. Isaac Gerald    male  43.0      1
405       Nourney, Mr. Alfred (Baron von Drachstedt")"    male  20.0      0
407                           Widener, Mr. George Dunton    male  50.0      1
411    Minahan, Mrs. William Edward (Lillian E Thorpe)  female  37.0      1
414                         Oliva y Ocana, Dona. Fermina  female  39.0      0

     Parch         Ticket      Fare            Cabin Embarked
12       0          21228   82.2667              B45        S
14       0    W.E.P. 5734   61.1750              E31        S
24       3       PC 17608  262.3750  B57 B59 B63 B66        C
26       1         113509   61.9792              B36        C
28       0         113054   30.5000              A21        S
..     ...            ...       ...              ...      ...
404      0          17765   27.7208              D40        C
405      0  SC/PARIS 2166   13.8625              D38        C
407      1         113503  211.5000              C80        C
411      0          19928   90.0000              C78        Q
414      0       PC 17758  108.9000             C105        C

[87 rows x 12 columns]
```

In [12]:
```python
df.duplicated()
```

Out[12]:
```
12     False
14     False
24     False
26     False
28     False
       ...
404    False
405    False
407    False
411    False
414    False
Length: 87, dtype: bool
```

In [13]:
```python
df.duplicated().sum()
```

Out[13]:
```
0
```

In [14]:
```python
df.describe()
```

Out[14]:

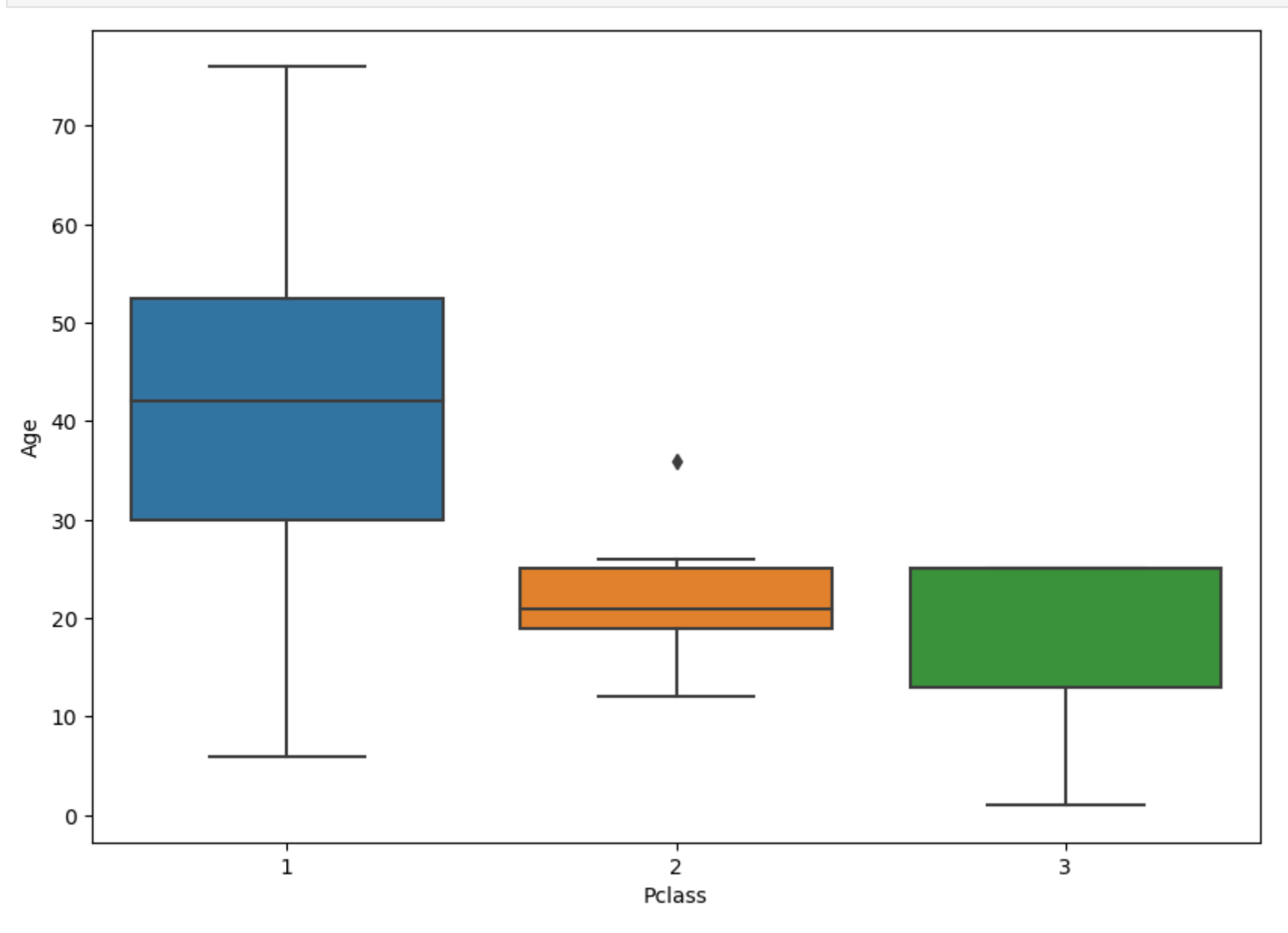
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	87.000000	87.000000	87.000000	87.000000	87.000000	87.000000	87.000000
mean	1102.712644	0.505747	1.137931	39.247126	0.597701	0.482759	98.109198
std	126.751901	0.502865	0.435954	15.218730	0.637214	0.860801	88.177319
min	904.000000	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000
25%	986.000000	0.000000	1.000000	27.000000	0.000000	0.000000	35.339600
50%	1094.000000	1.000000	1.000000	39.000000	1.000000	0.000000	71.283300
75%	1216.000000	1.000000	1.000000	50.000000	1.000000	1.000000	135.066650
max	1306.000000	1.000000	3.000000	76.000000	3.000000	4.000000	512.329200

In [15]:
```python
df.skew()
```

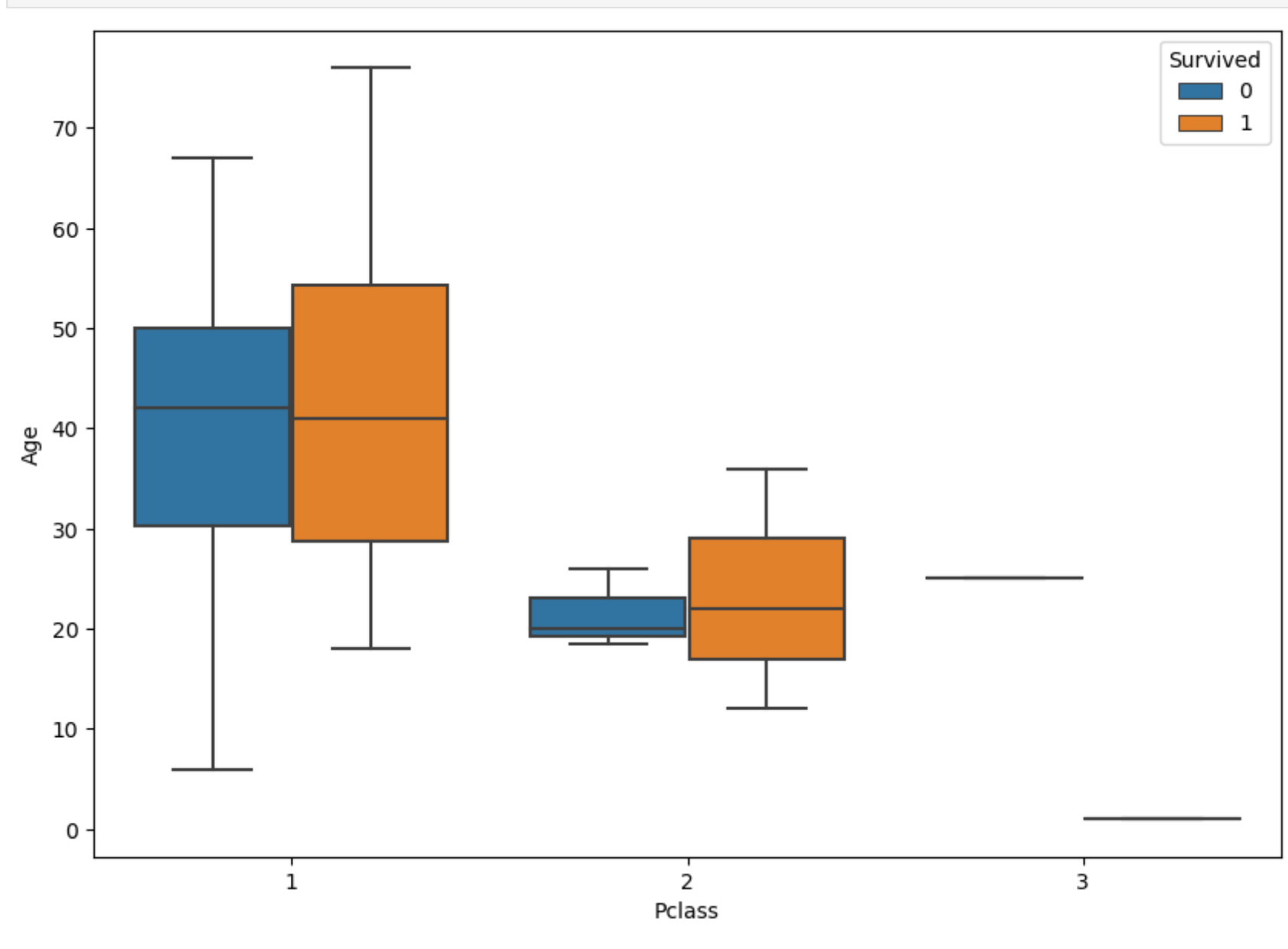
```
C:\Users\Sutharsahana\AppData\Local\Temp\ipykernel_7460\1665899112.py:1: FutureWarning: The default value of numeric_only in DataFrame.skew is deprecated. In a future versio
n, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warnin
g.
  df.skew()
```

Out[15]:
```
PassengerId    0.088289
Survived      -0.023395
Pclass         3.304943
Age           -0.026420
SibSp          0.865184
Parch          2.013732
Fare           1.790422
dtype: float64
```

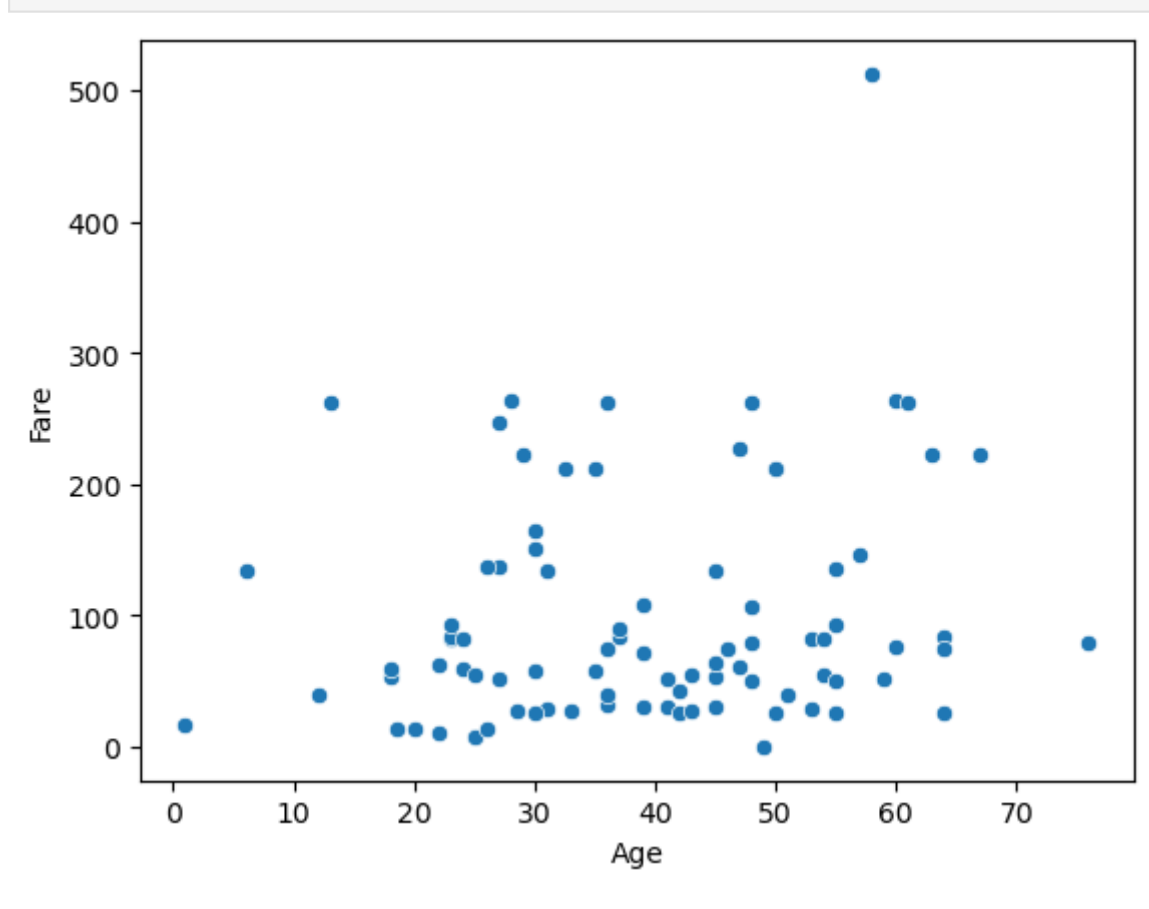
In [17]:
```python
plt.figure(figsize=(10,7))
sns.boxplot(data=df, y='Age', x='Pclass')
plt.show()
```



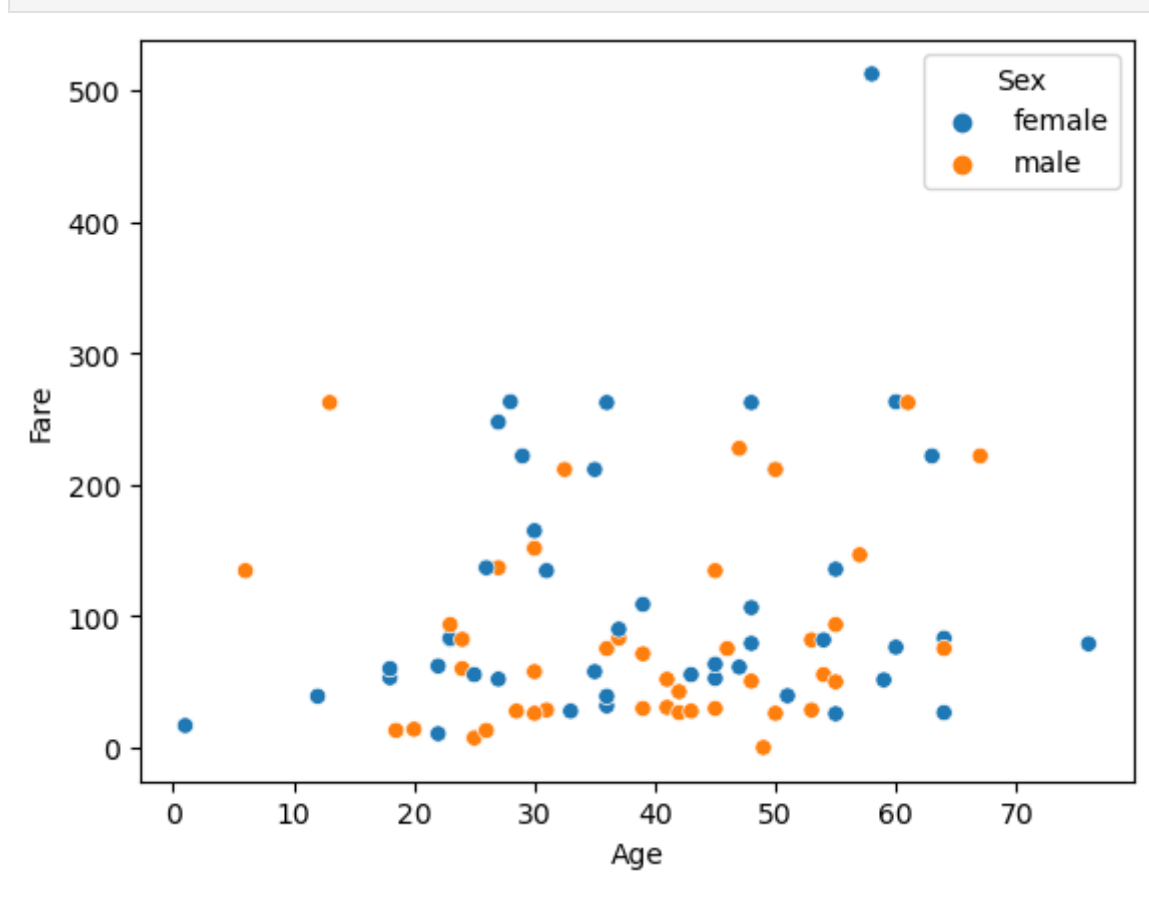
In [18]:
```python
plt.figure(figsize=(10,7))
sns.boxplot(data=df, y='Age', x='Pclass', hue='Survived')
plt.show()
```



In [19]:
```python
sns.scatterplot(x=df['Age'], y=df['Fare'])
plt.show()
```



In [20]:
```python
sns.scatterplot(data=df, x='Age', y='Fare', hue='Sex')
plt.show()
```

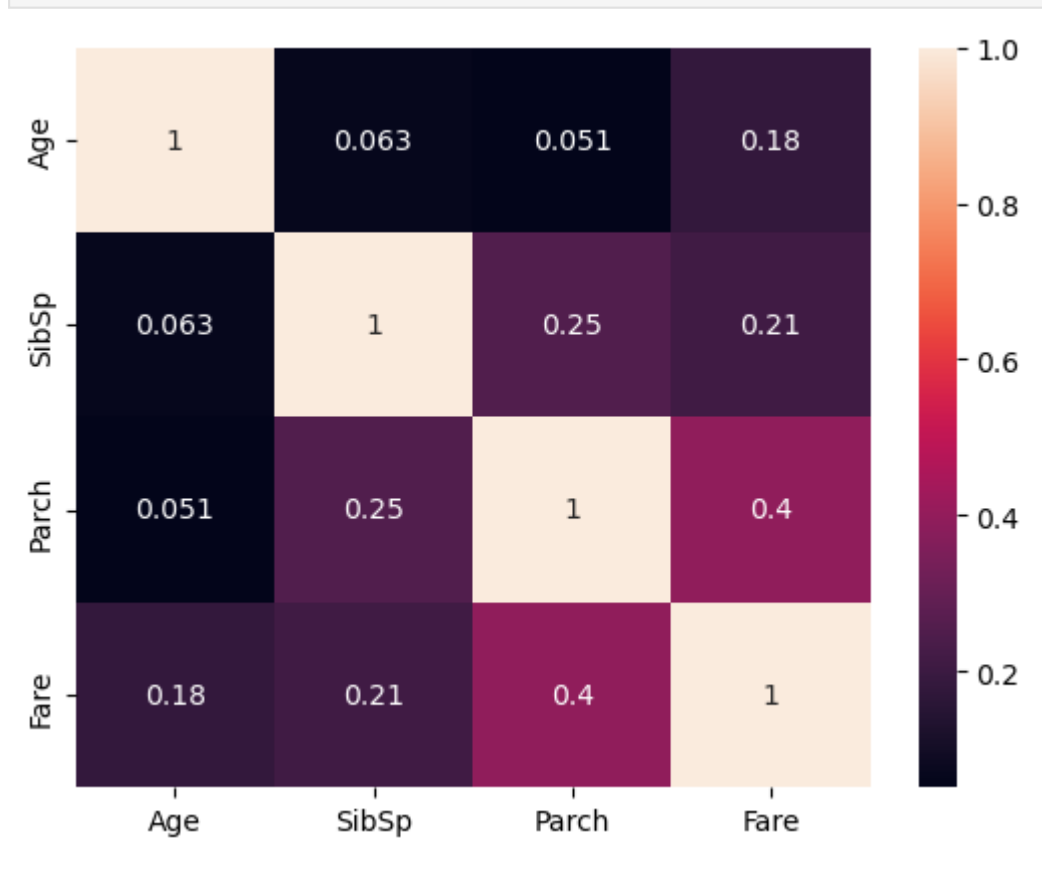


In [23]:
```python
correlation_matrix = df[['Age','SibSp','Parch','Fare']].corr()
correlation_matrix
```

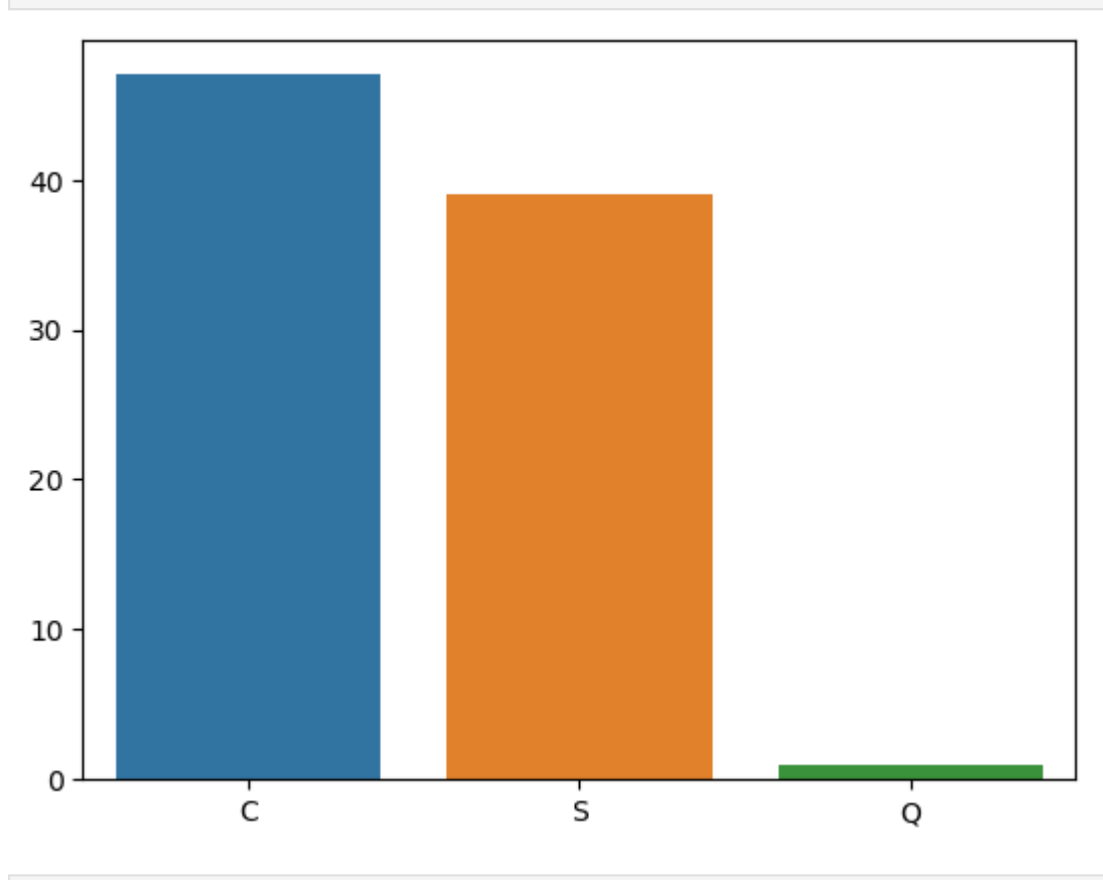
Out[23]:

	Age	SibSp	Parch	Fare
Age	1.000000	0.062530	0.051144	0.180567
SibSp	0.062530	1.000000	0.252194	0.213014
Parch	0.051144	0.252194	1.000000	0.395685
Fare	0.180567	0.213014	0.395685	1.000000

In [24]:
```python
sns.heatmap(correlation_matrix, annot=True)
plt.show()
```

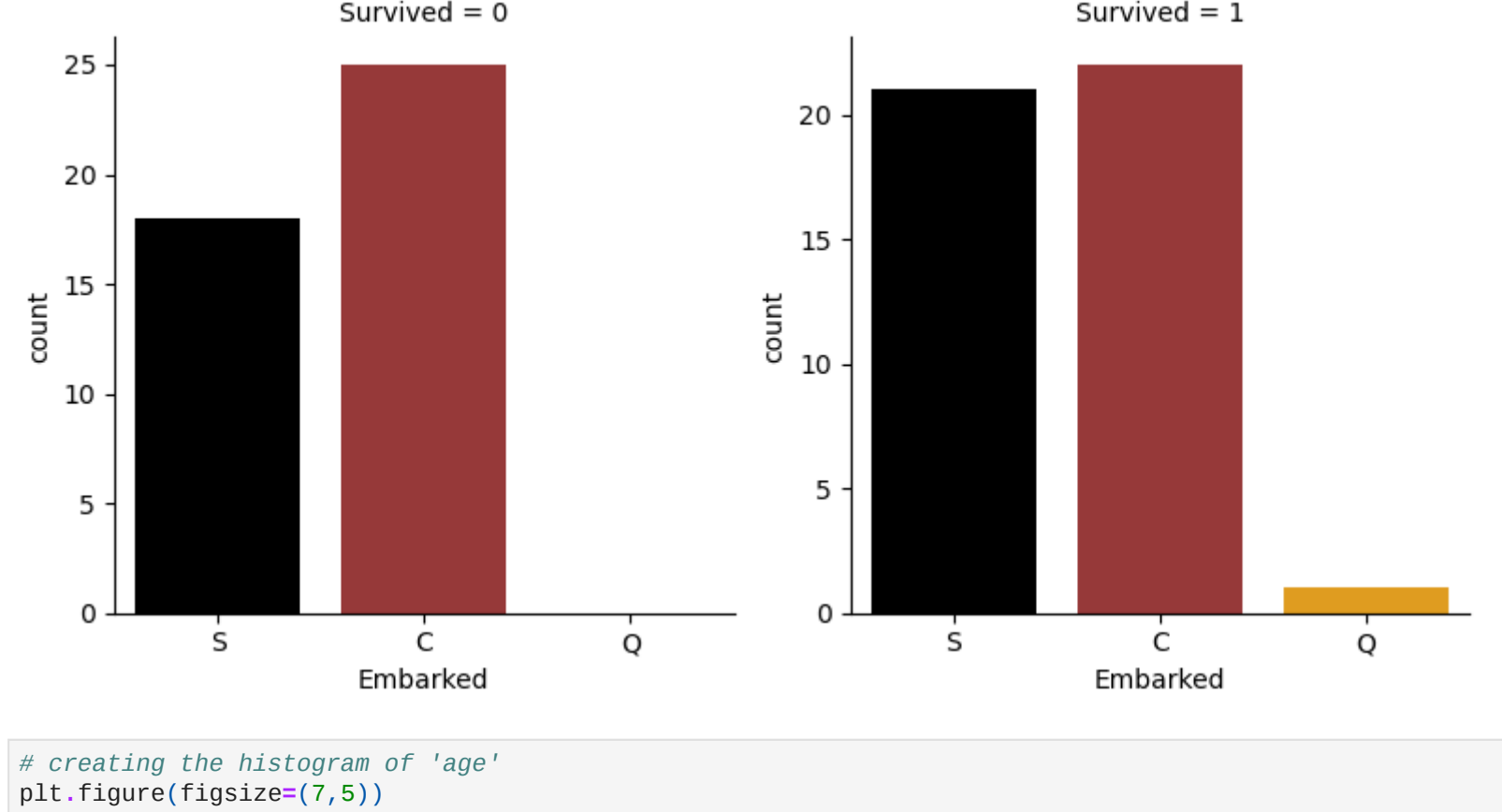


In [30]:
```python
sns.barplot(x=df['Embarked'].value_counts().index,
            y=df['Embarked'].value_counts().values)
plt.show()
```

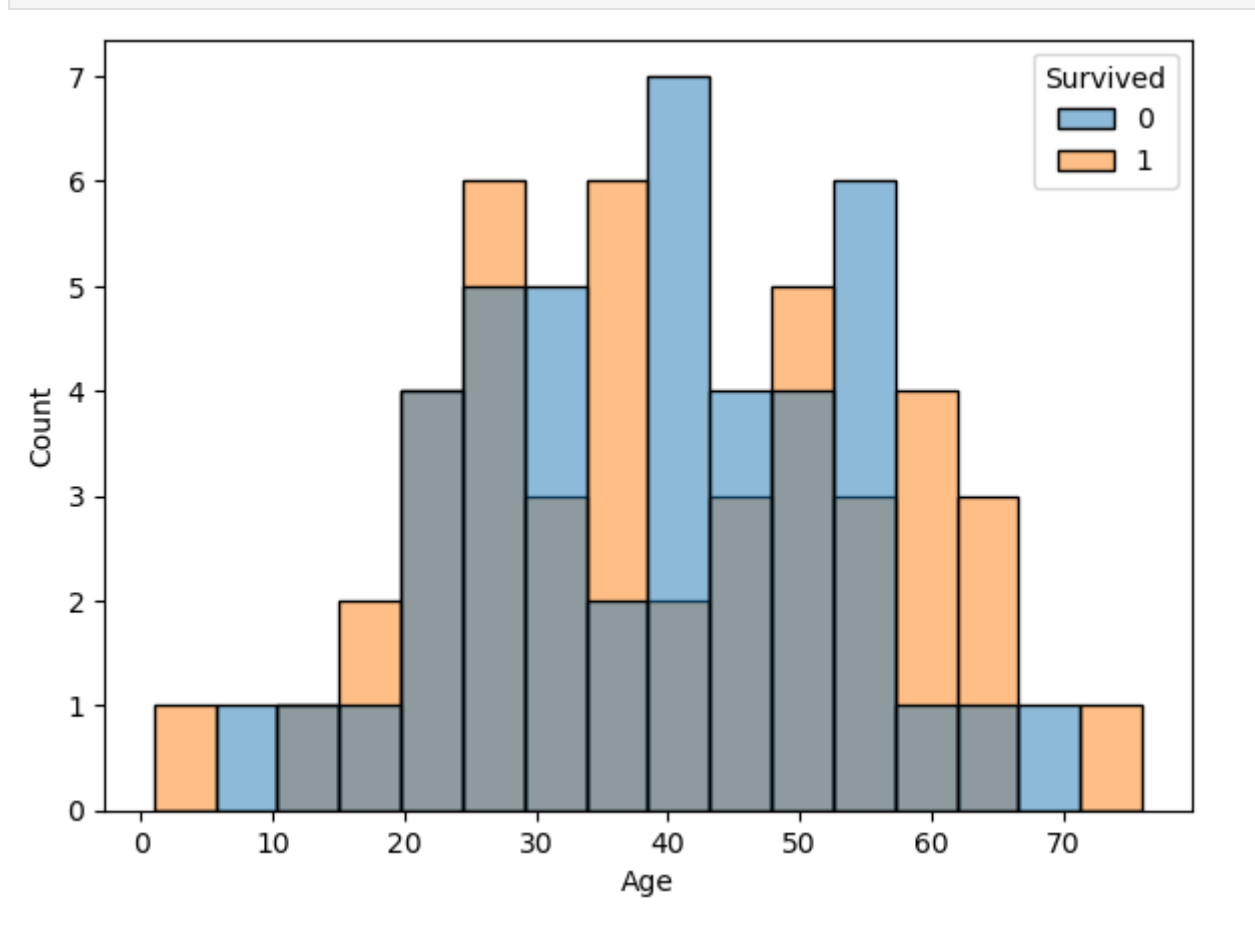


In [31]:
```python
# creating a facet grid with columns as survived=0 and survived=1
grid = sns.FacetGrid(data=df, col='Survived', height=4, aspect=1, sharey=False)
# mapping bar plot and the data on to the grid
grid.map(sns.countplot, 'Embarked', palette=['black', 'brown', 'orange'])
plt.show()
```

```
C:\Users\Sutharsahana\.ms-ad\lib\site-packages\seaborn\axisgrid.py:712: UserWarning: Using the countplot function without specifying `order` is likely to produce an incorrec
t plot.
  warnings.warn(warning)
```



In [32]:
```python
# creating the histogram of 'age'
plt.figure(figsize=(7,5))
sns.histplot(data=df, x='Age', bins=16, hue='Survived')
plt.show()
```

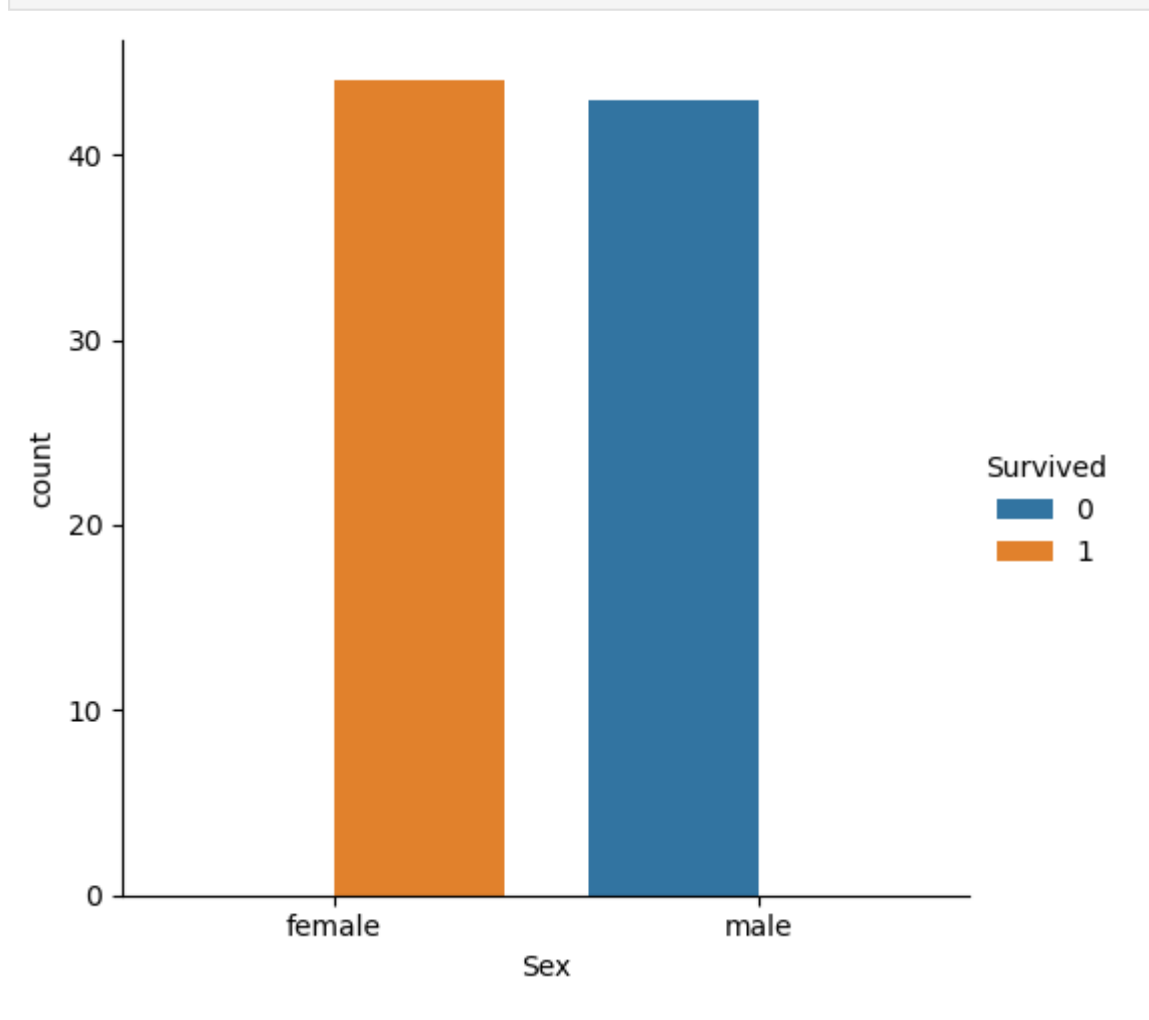


In [34]:
```python
#1.1
import seaborn as sns

# Assuming you have a DataFrame called 'df' with variables 'variable1' and 'variable2'
#df = pd.DataFrame({'variable1': ['A', 'B', 'A', 'B', 'A'],
                  #'variable2': ['X', 'Y', 'X', 'Y', 'Z']})

# Create a countplot with hue
sns.catplot(x='Sex', hue='Survived', data=df, kind='count')

# Display the plot
plt.show()
```

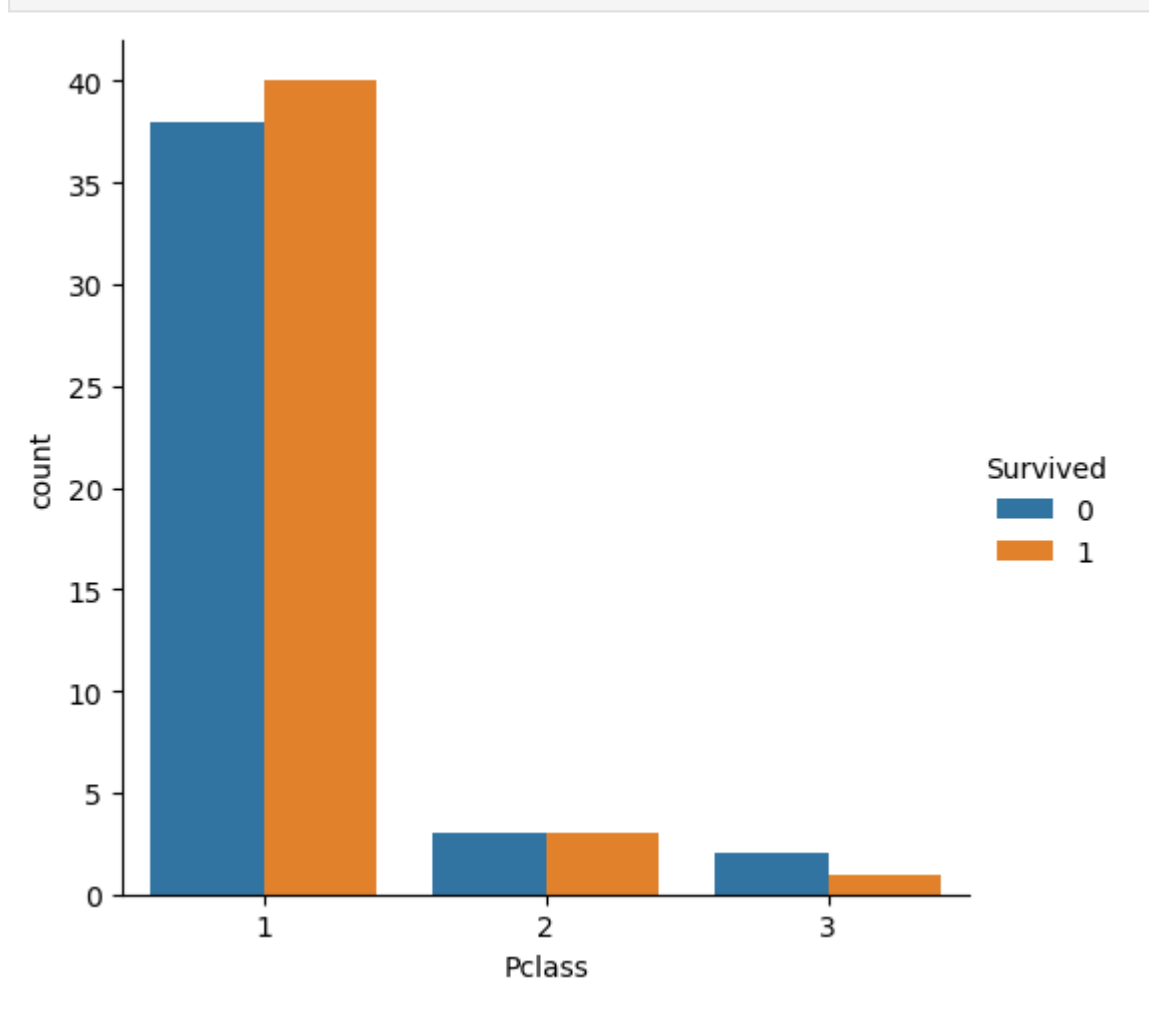


In [38]:
```python
#1.3
import seaborn as sns

# Assuming you have a DataFrame called 'df' with variables 'variable1' and 'variable2'
#df = pd.DataFrame({'variable1': ['A', 'B', 'A', 'B', 'A'],
                  #'variable2': ['X', 'Y', 'X', 'Y', 'Z']})

# Create a countplot with hue
sns.catplot(x='Pclass', hue='Survived', data=df, kind='count')

# Display the plot
plt.show()
```



In [ ]: