In [ ]:
```
# importing libraries
```

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

In [ ]:
```
# importing datasets
```

In [3]:
```
titanic = pd.read_csv('Titanic_data.csv')
titanic
```

Out[3]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
...	...	...	...	...	...	...	...	...	...
884	887	0	2	Male	27.0	0	0	13.000	3
885	888	1	1	female	19.0	0	0	30.000	3
886	889	0	3	female	60.0	1	2	23.450	3
887	890	1	1	Male	26.0	0	0	30.000	1
888	891	0	3	Male	32.0	0	0	7.750	2

889 rows × 9 columns

In [ ]:
```
# data exploration
```

In [5]:
```
titanic.head()
```

Out[5]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3

In [6]:
```
# to check first 10 rows in the table
titanic.head(10)
```

Out[6]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
5	6	0	3	Male	60.0	0	0	8.458	2
6	7	0	1	Male	54.0	0	0	51.863	3
7	8	0	3	Male	2.0	3	1	21.075	3
8	9	1	3	female	27.0	0	2	11.133	3
9	10	1	2	female	14.0	1	0	30.071	1

In [7]:
```
titanic.head(20)
```

Out[7]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
5	6	0	3	Male	60.0	0	0	8.458	2
6	7	0	1	Male	54.0	0	0	51.863	3
7	8	0	3	Male	2.0	3	1	21.075	3
8	9	1	3	female	27.0	0	2	11.133	3
9	10	1	2	female	14.0	1	0	30.071	1
10	11	1	3	female	4.0	1	1	16.700	3
11	12	1	1	female	58.0	0	0	26.550	3
12	13	0	3	Male	20.0	0	0	8.050	3
13	14	0	3	Male	39.0	1	5	31.275	3
14	15	0	3	female	14.0	0	0	7.854	3
15	16	1	2	female	55.0	0	0	16.000	3
16	17	0	3	Male	2.0	4	1	29.125	2
17	18	1	2	Male	60.0	0	0	13.000	3
18	19	0	3	female	31.0	1	0	18.000	3
19	20	1	3	female	60.0	0	0	7.225	1

In [11]:
```
# assign variables to the 1st 10 and 1st 20 rows

abc = titanic.head(10)
abc
```

Out[11]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
5	6	0	3	Male	60.0	0	0	8.458	2
6	7	0	1	Male	54.0	0	0	51.863	3
7	8	0	3	Male	2.0	3	1	21.075	3
8	9	1	3	female	27.0	0	2	11.133	3
9	10	1	2	female	14.0	1	0	30.071	1

In [17]:
```
d = titanic.head(20)
d
```

Out[17]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
5	6	0	3	Male	60.0	0	0	8.458	2
6	7	0	1	Male	54.0	0	0	51.863	3
7	8	0	3	Male	2.0	3	1	21.075	3
8	9	1	3	female	27.0	0	2	11.133	3
9	10	1	2	female	14.0	1	0	30.071	1
10	11	1	3	female	4.0	1	1	16.700	3
11	12	1	1	female	58.0	0	0	26.550	3
12	13	0	3	Male	20.0	0	0	8.050	3
13	14	0	3	Male	39.0	1	5	31.275	3
14	15	0	3	female	14.0	0	0	7.854	3
15	16	1	2	female	55.0	0	0	16.000	3
16	17	0	3	Male	2.0	4	1	29.125	2
17	18	1	2	Male	60.0	0	0	13.000	3
18	19	0	3	female	31.0	1	0	18.000	3
19	20	1	3	female	60.0	0	0	7.225	1

In [18]:
```
# to check the number of rows and columns in the dataset

titanic.shape
```

Out[18]:
```
(889, 9)
```

In [19]:
```
# to check the information of the dataset

titanic.info
```

Out[19]:
```
<bound method DataFrame.info of      Passenger_ID  Survived  Passenger_class     Sex   Age  Number_SP  \
0               1         0                3    Male  22.0          1
1               2         1                1  female  38.0          1
2               3         1                3  female  26.0          0
3               4         1                1  female  35.0          1
4               5         0                3    Male  35.0          0
..            ...       ...              ...     ...   ...        ...
884           887         0                2    Male  27.0          0
885           888         1                1  female  19.0          0
886           889         0                3  female  60.0          1
887           890         1                1    Male  26.0          0
888           891         0                3    Male  32.0          0

     Number_PC    Fare  Embarked
0            0   7.250         3
1            0  71.283         1
2            0   7.925         3
3            0  53.100         3
4            0   8.050         3
..         ...     ...       ...
884          0  13.000         3
885          0  30.000         3
886          2  23.450         3
887          0  30.000         1
888          0   7.750         2

[889 rows x 9 columns]>
```

In [ ]:
```
# basic statistics
```

In [21]:
```
# to show statistical values of the dataset

titanic.describe()
```

Out[21]:

	Passenger_ID	Survived	Passenger_class	Age	Number_SP	Number_PC	Fare	Embarked
count	889.000000	889.000000	889.000000	889.000000	889.000000	889.000000	889.000000	889.000000
mean	446.000000	0.382452	2.311586	35.686355	0.524184	0.382452	32.096711	2.535433
std	256.998173	0.486260	0.834700	17.756733	1.103705	0.806761	49.697497	0.792088
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000	1.000000
25%	224.000000	0.000000	2.000000	22.000000	0.000000	0.000000	7.896000	2.000000
50%	446.000000	0.000000	3.000000	32.000000	0.000000	0.000000	14.454000	3.000000
75%	668.000000	1.000000	3.000000	54.000000	1.000000	0.000000	31.000000	3.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329000	3.000000

In [22]:
```
# to show statistical values of age and fare

titanic[['Age','Fare']].describe()
```

Out[22]:

	Age	Fare
count	889.000000	889.000000
mean	35.686355	32.096711
std	17.756733	49.697497
min	0.420000	0.000000
25%	22.000000	7.896000
50%	32.000000	14.454000
75%	54.000000	31.000000
max	80.000000	512.329000

In [23]:
```
# to check average fare

titanic[['Fare']].mean()
```

Out[23]:
```
Fare    32.096711
dtype: float64
```

In [25]:
```
# sorting and filtering

# to check for null values
```

In [33]:
```
titanic.isnull().sum()
```

Out[33]:
```
Passenger_ID       0
Survived           0
Passenger_class    0
Sex                0
Age                0
Number_SP          0
Number_PC          0
Fare               0
Embarked           0
dtype: int64
```

In [34]:
```
titanic.isnull().sum().sum()
```

Out[34]:
```
0
```

In [35]:
```
# to check the length of the dataset

len(titanic)
```

Out[35]:
```
889
```

In [36]:
```
# to filter out passengers in 1st class
```

In [39]:
```
Firstclass_P = titanic[titanic['Passenger_class'] ==1]
```

In [40]:
```
Firstclass_P
```

Out[40]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
1	2	1	1	female	38.0	1	0	71.283	1
3	4	1	1	female	35.0	1	0	53.100	3
6	7	0	1	Male	54.0	0	0	51.863	3
11	12	1	1	female	58.0	0	0	26.550	3
23	24	1	1	Male	28.0	0	0	35.500	3
...	...	...	...	...	...	...	...	...	...
869	872	1	1	female	47.0	1	1	52.554	3
870	873	0	1	Male	33.0	0	0	5.000	3
877	880	1	1	female	56.0	0	1	83.158	1
885	888	1	1	female	19.0	0	0	30.000	3
887	890	1	1	Male	26.0	0	0	30.000	1

214 rows × 9 columns

In [49]:
```
Adults = titanic[titanic['Age'] > 18.0]
```

In [50]:
```
Adults
```

Out[50]:

	Passenger_ID	Survived	Passenger_class	Sex	Age	Number_SP	Number_PC	Fare	Embarked
0	1	0	3	Male	22.0	1	0	7.250	3
1	2	1	1	female	38.0	1	0	71.283	1
2	3	1	3	female	26.0	0	0	7.925	3
3	4	1	1	female	35.0	1	0	53.100	3
4	5	0	3	Male	35.0	0	0	8.050	3
...	...	...	...	...	...	...	...	...	...
884	887	0	2	Male	27.0	0	0	13.000	3
885	888	1	1	female	19.0	0	0	30.000	3
886	889	0	3	female	60.0	1	2	23.450	3
887	890	1	1	Male	26.0	0	0	30.000	1
888	891	0	3	Male	32.0	0	0	7.750	2

750 rows × 9 columns

In [ ]:
```
     #Data analysis and Visualization
```

In [51]:
```
# to check the number of passengers who died(i.e 0) and number of passengers who survived(1)
```

In [52]:
```
survival_counts = titanic['Survived'].value_counts()
print('Total number of passengers who died:', survival_counts[0])
print('Total number of passengers who survived:', survival_counts[1])
```

```
Total number of passengers who died: 549
Total number of passengers who survived: 340
```

In [53]:
```
# to check the number of survival by Sex
```

In [54]:
```
Survival_by_Sex =titanic.groupby(['Sex', 'Survived']).agg({'Survived':'count'})
print(Survival_by_Sex)
```

```
                 Survived
Sex    Survived
Male   0              468
       1              109
female 0               81
       1              231
```

In [55]:
```
# to check the number of survival by Age
Survival_by_Age =titanic.groupby(['Age', 'Survived']).agg({'Survived':'count'})
print(Survival_by_Age)
```

```
               Survived
Age   Survived
0.42  1               1
0.67  1               1
0.75  1               2
0.83  1               2
0.92  1               1
...                 ...
70.00 0               2
70.50 0               1
71.00 0               2
74.00 0               1
80.00 1               1

[142 rows x 1 columns]
```

In [84]:
```
# Survival rate of Parents to children
Survival_PC =titanic.groupby(['Number_PC', 'Survived']).agg({'Survived':'count'})
print(Survival_PC)
```

```
                    Survived
Number_PC Survived
0         0              445
          1              231
1         0               53
          1               65
2         0               40
          1               40
3         0                2
          1                3
4         0                4
5         0                4
          1                1
6         0                1
```

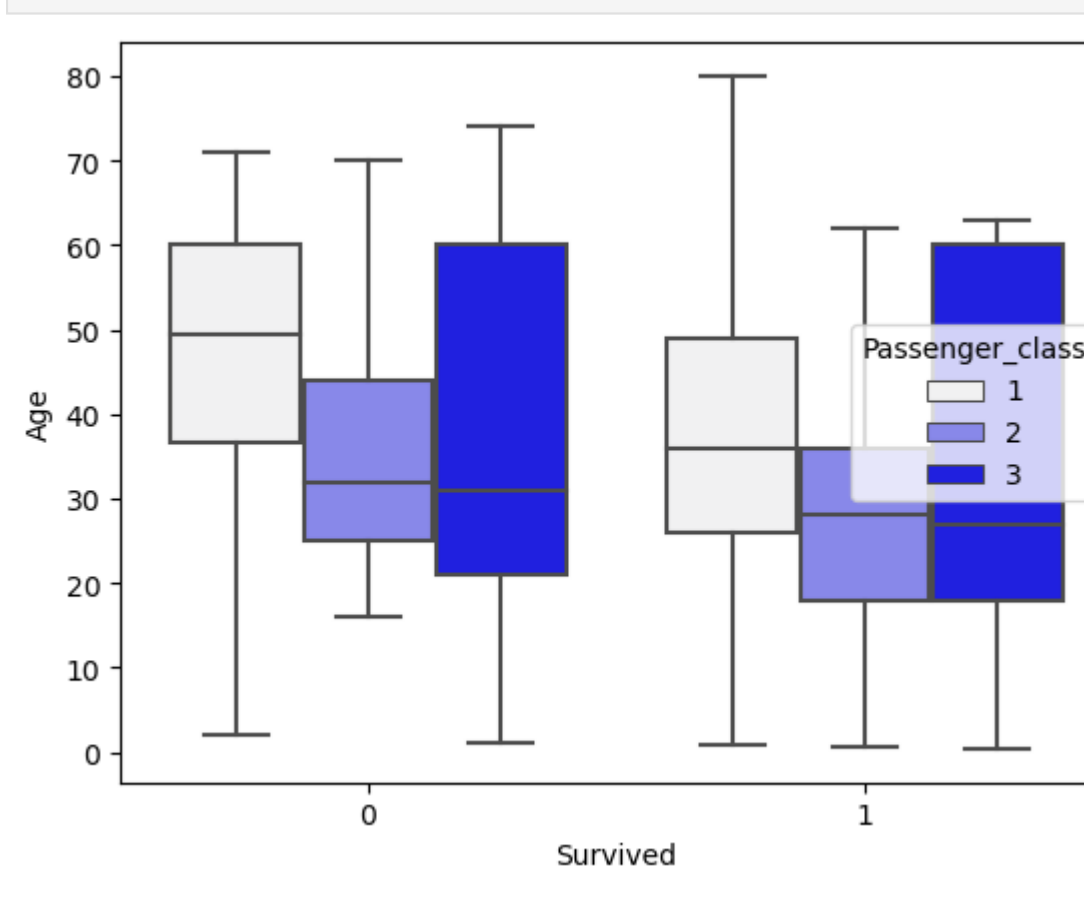
In [57]:
```
# Percentage of survival and death rates
Survival_rate = titanic['Survived'].value_counts(normalize=True)
print('Percentage of those who survived:', Survival_rate[1]*100, '%')
print('Percentage of those who died:', Survival_rate[0]*100, '%')
```

```
Percentage of those who survived: 38.24521934758155 %
Percentage of those who died: 61.754780652418454 %
```

In [58]:
```
# number of those who

# Survival rate of Siblings to Spouse
Survival_SP =titanic.groupby(['Number_SP', 'Survived']).agg({'Survived':'count'})
print(Survival_SP)
```

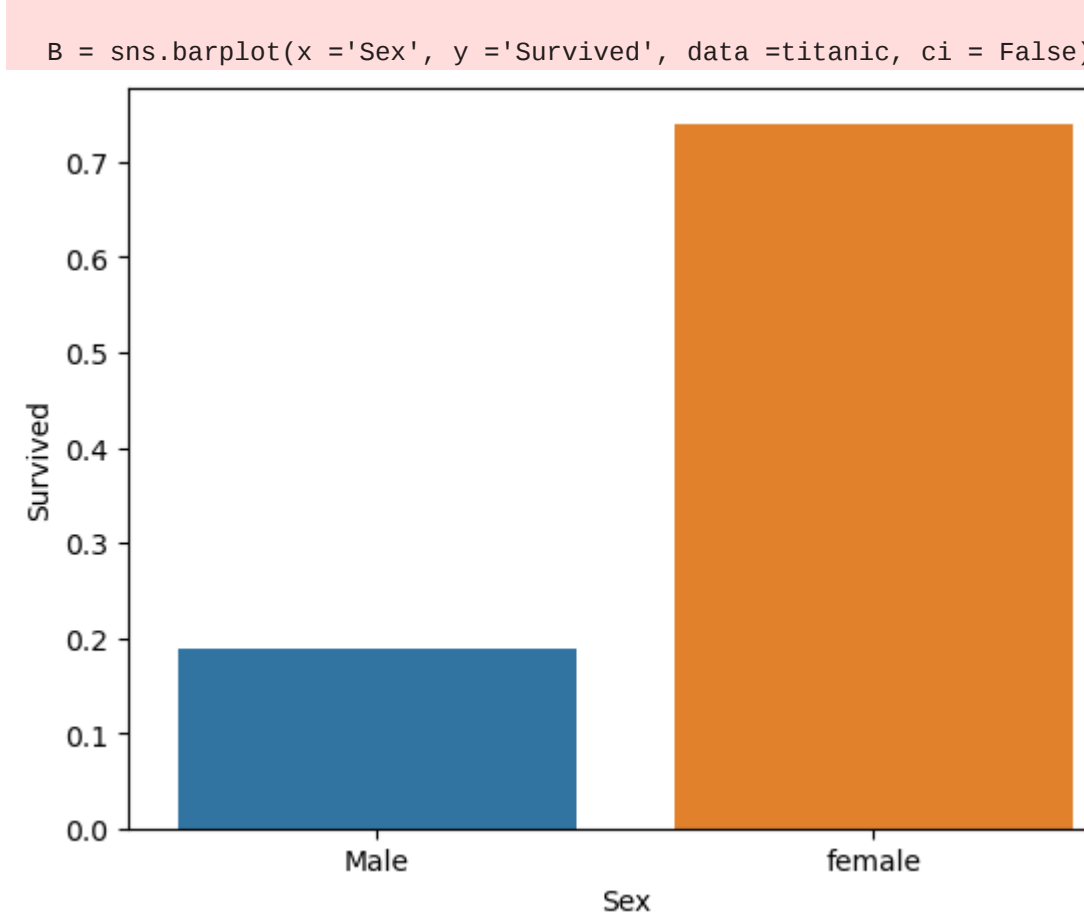
```
                    Survived
Number_SP Survived
0         0              398
          1              208
1         0               97
          1              112
2         0               15
          1               13
3         0               12
          1                4
4         0               15
          1                3
5         0                5
8         0                7
```

In [94]:
```
x = sns.boxplot(x = 'Survived', y = 'Age',
               data = titanic, color = 'blue', hue ='Passenger_class')
```



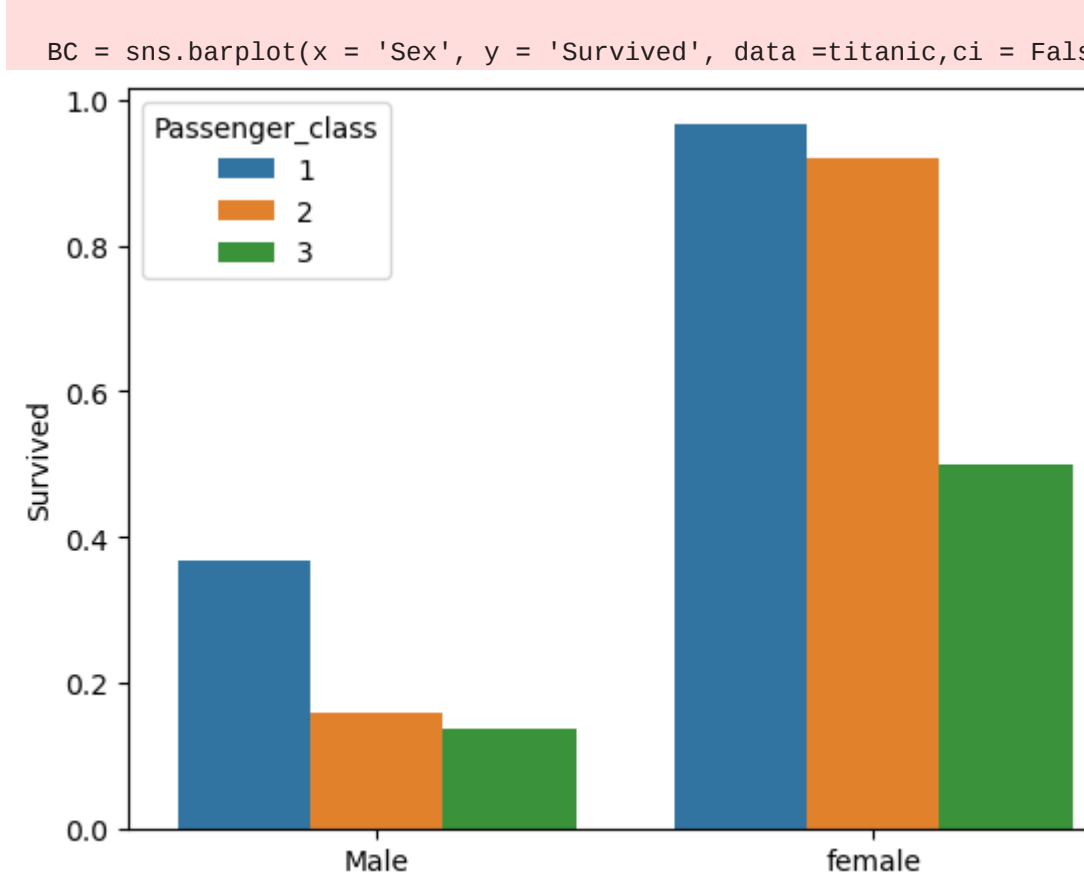
In [81]:
```
# to show survival  rate by Sex

B = sns.barplot(x ='Sex', y ='Survived', data =titanic, ci = False)
```

```
C:\Users\JOSEPH MARTHA\AppData\Local\Temp\ipykernel_1356\2129231748.py:3: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', False)` for the same effect.

  B = sns.barplot(x ='Sex', y ='Survived', data =titanic, ci = False)
```



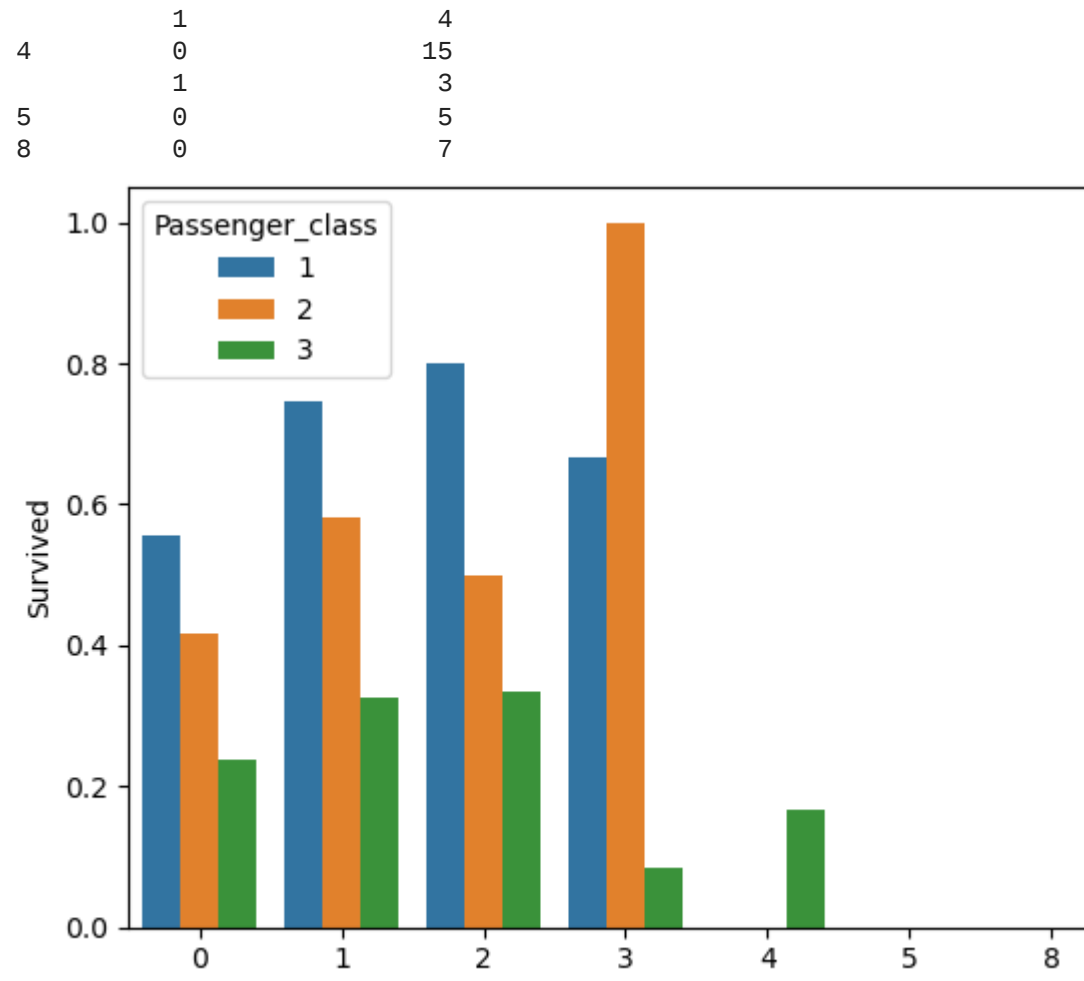
In [82]:
```
# to show survival rate by sex and passebger class
BC = sns.barplot(x = 'Sex', y = 'Survived', data =titanic,ci = False, hue = 'Passenger_class')
```

```
C:\Users\JOSEPH MARTHA\AppData\Local\Temp\ipykernel_1356\452776273.py:2: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', False)` for the same effect.

  BC = sns.barplot(x = 'Sex', y = 'Survived', data =titanic,ci = False, hue = 'Passenger_class')
```



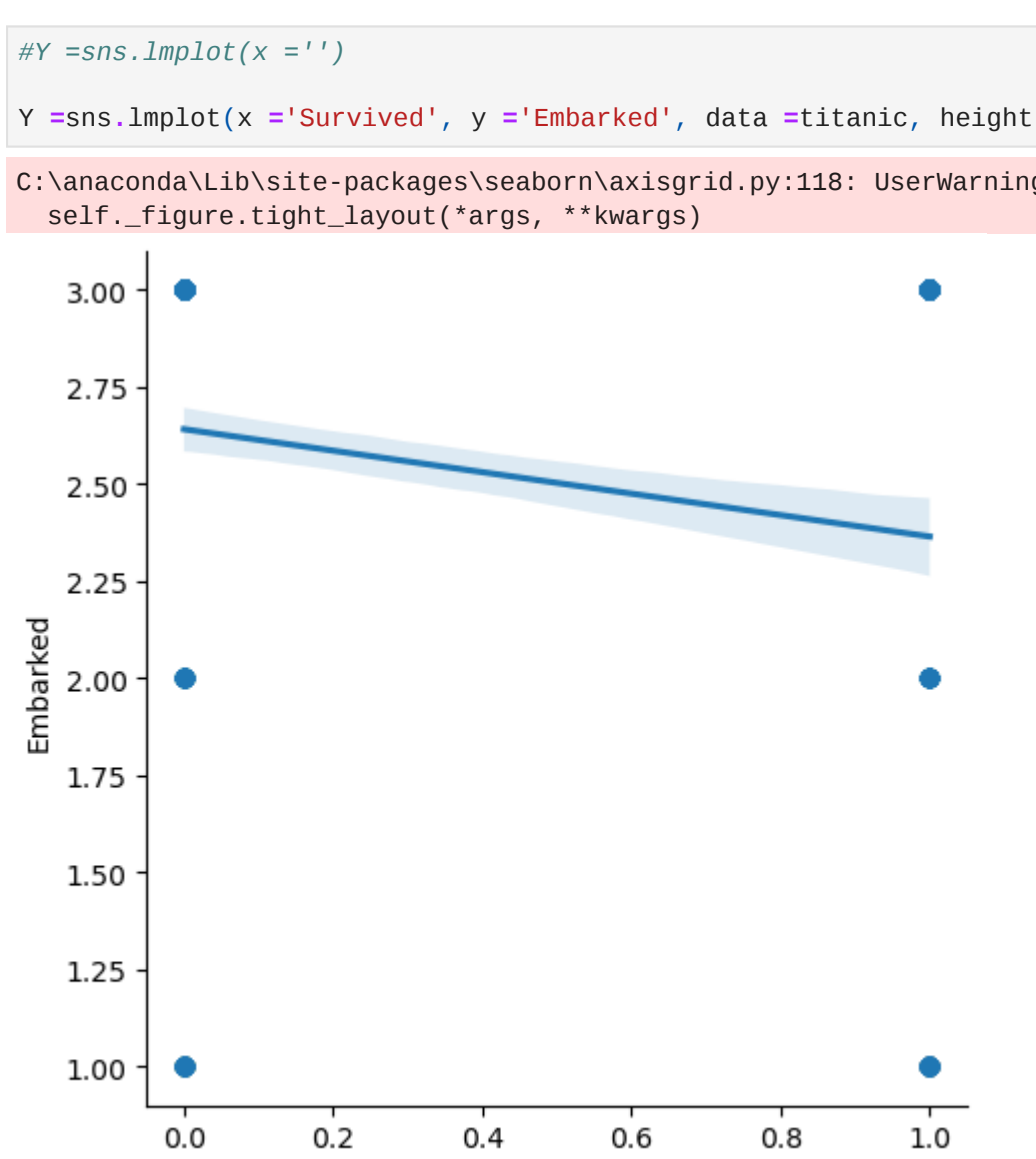
In [86]:
```
# sU
PC = sns.barplot(x = 'Number_SP', y = 'Survived', data =titanic,ci = False, hue = 'Passenger_class')
Survival_SP =titanic.groupby(['Number_SP', 'Survived']).agg({'Survived':'count'})
print(Survival_SP)
```

```
C:\Users\JOSEPH MARTHA\AppData\Local\Temp\ipykernel_1356\1353911597.py:2: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', False)` for the same effect.

  PC = sns.barplot(x = 'Number_SP', y = 'Survived', data =titanic,ci = False, hue = 'Passenger_class')
                    Survived
Number_SP Survived
0         0              398
          1              208
1         0               97
          1              112
2         0               15
          1               13
3         0               12
          1                4
4         0               15
          1                3
5         0                5
8         0                7
```



In [90]:
```
#Y =sns.lmplot(x ='')

Y =sns.lmplot(x ='Survived', y ='Embarked', data =titanic, height = 5)
```

```
C:\anaconda\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



In [1]:

In [ ]: