In [1]:

```
import pandas as pd
df=pd.read_csv("train.csv")
df.head(3)
```

Out[1]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250

In [2]:

```
df.describe(include='all')
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Par
count	891.000000	891.000000	891.000000	891	891	714.000000	891.000000	891.0000
unique	NaN	NaN	NaN	891	2	NaN	NaN	N
top	NaN	NaN	NaN	Braund, Mr. Owen Harris	male	NaN	NaN	N
freq	NaN	NaN	NaN	1	577	NaN	NaN	N
mean	446.000000	0.383838	2.308642	NaN	NaN	29.699118	0.523008	0.3815
std	257.353842	0.486592	0.836071	NaN	NaN	14.526497	1.102743	0.8060
min	1.000000	0.000000	1.000000	NaN	NaN	0.420000	0.000000	0.0000
25%	223.500000	0.000000	2.000000	NaN	NaN	20.125000	0.000000	0.0000
50%	446.000000	0.000000	3.000000	NaN	NaN	28.000000	0.000000	0.0000
75%	668.500000	1.000000	3.000000	NaN	NaN	38.000000	1.000000	0.0000
max	891.000000	1.000000	3.000000	NaN	NaN	80.000000	8.000000	6.0000

DATA FILTERING

In [3]:

```
df.columns
```

Out[3]:

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

In [5]:
```
df[['Name','Age','Fare']]  #if you ant to mention more than 2 cols, put 2 sq
```

Out[5]:

	Name	Age	Fare
0	Braund, Mr. Owen Harris	22.0	7.2500
1	Cumings, Mrs. John Bradley (Florence Briggs Th...	38.0	71.2833
2	Heikkinen, Miss. Laina	26.0	7.9250
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.0	53.1000
4	Allen, Mr. William Henry	35.0	8.0500
...	...	...	...
886	Montvila, Rev. Juozas	27.0	13.0000
887	Graham, Miss. Margaret Edith	19.0	30.0000
888	Johnston, Miss. Catherine Helen "Carrie"	NaN	23.4500
889	Behr, Mr. Karl Howell	26.0	30.0000
890	Dooley, Mr. Patrick	32.0	7.7500

891 rows × 3 columns

In [6]:
```
sum(df['Sex']=='male') #TOTAL NO. OF MALES
```

Out[6]:
```
577
```

In [7]:
```
df[df['Sex']=='male']  #TO EXTRACT ALL INFO ABOUT MALE TRAVELER
```

Out[7]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Far
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.250
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.050
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.458
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.862
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.075
...	...	...	...	...	...	...	...	...	...	.
883	884	0	2	Banfield, Mr. Frederick James	male	28.0	0	0	C.A./SOTON 34068	10.500
884	885	0	3	Sutehall, Mr. Henry Jr	male	25.0	0	0	SOTON/OQ 392076	7.050
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.000
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.000
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.750

577 rows × 12 columns

In [9]:
```
#NO. OF SURVIVORS
sum(df['Survived']==1)
```

Out[9]:
```
342
```

In [11]:
```
sum((df['Survived']==1) & (df['Sex']=="female")) #NO. OF FEMALE SURVIVORS
```

Out[11]:
```
233
```

CHECK NULL VALUES

In [12]:

```
df.isnull().sum()
```

Out[12]:

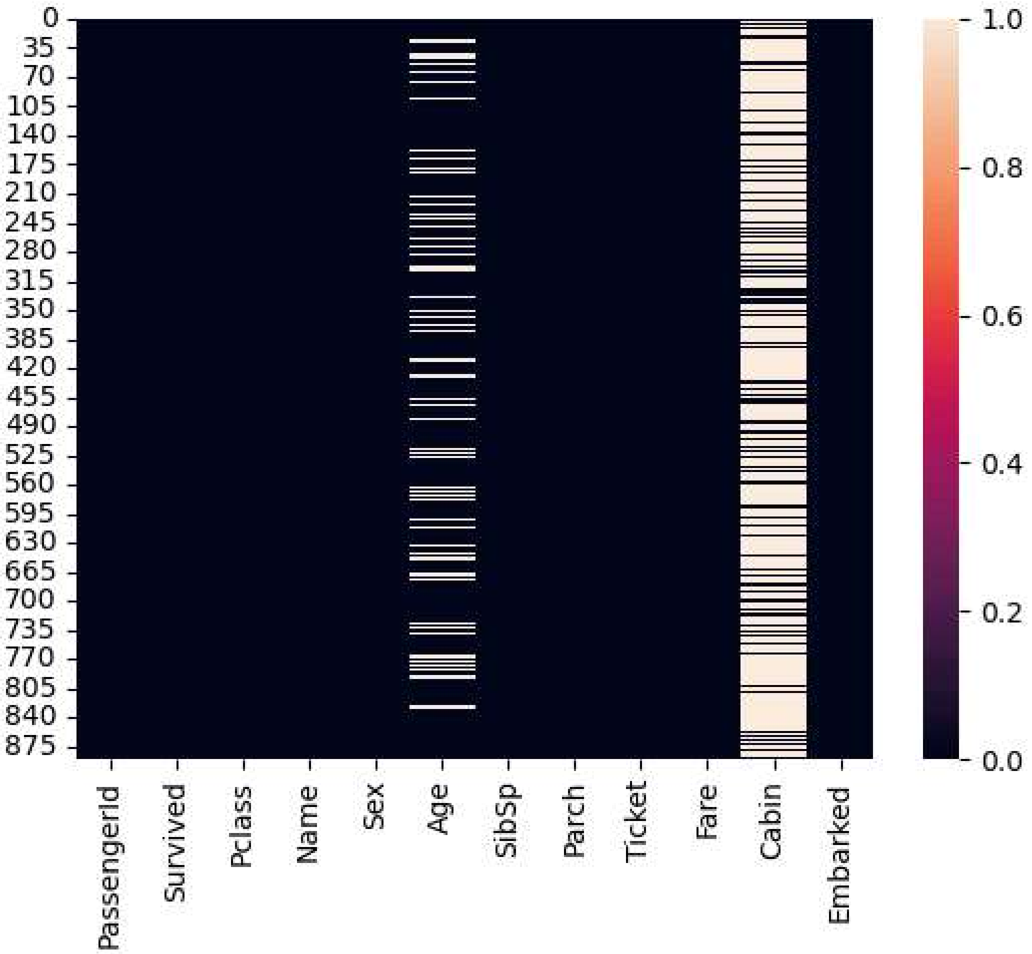
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

In [13]:

```
import seaborn as sns
sns.heatmap(df.isnull())
```

Out[13]:

```
<AxesSubplot:>
```



In [16]:
```
percentage_missing=df.isnull().sum()*100/len(df) #Percentage of missing valu
percentage_missing
```

Out[16]:
```
PassengerId     0.000000
Survived        0.000000
Pclass          0.000000
Name            0.000000
Sex             0.000000
Age            19.865320
SibSp           0.000000
Parch           0.000000
Ticket          0.000000
Fare            0.000000
Cabin          77.104377
Embarked        0.224467
dtype: float64
```

In [19]:
```
df.drop(['Cabin'], axis=1,inplace=True)
```

In [20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 11 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  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(4)
memory usage: 76.7+ KB
```

HANDLE MISSING VALUES

In [22]:
```
df['Embarked'].mode()
```

Out[22]:
```
0    S
Name: Embarked, dtype: object
```

In [23]:
```
df['Embarked'].fillna('S',inplace=True)
```

In [24]:
```
df.isnull().sum()
```

Out[24]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Embarked         0
dtype: int64
```

In [25]:
```
df['Age'].mean()
```

Out[25]:
```
29.69911764705882
```

In [26]:
```
df['Age'].fillna(29, inplace=True)
```

In [27]:
```
df.isnull().sum()
```

Out[27]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

CATEGORICAL DATA ENCODING

In [28]:
```
df['Sex'].unique()
```

Out[28]:
```
array(['male', 'female'], dtype=object)
```

In [29]:
```
df['Gender']=df['Sex'].map({'male':1,'female':0})
```

In [30]:
```
df.head(1)
```

Out[30]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25	S

In [32]:
```
x=df['Sex'].map({'male':1,'female':0}) #2nd method
```

In [33]:
```
df.insert(5,'Gender_New',x)
df.head(1)
```

Out[33]:

	PassengerId	Survived	Pclass	Name	Sex	Gender_New	Age	SibSp	Parch	Ticket	Fa
0	1	0	3	Braund, Mr. Owen Harris	male	1	22.0	1	0	A/5 21171	7.

In [35]:
```
df['Embarked'].unique()
```

Out[35]:
```
array(['S', 'C', 'Q'], dtype=object)
```

```
In [36]: pd.get_dummies(df,columns=['Embarked']) #from one dummy col, we can predict
```

Out[36]:

	PassengerId	Survived	Pclass	Name	Sex	Gender_New	Age	SibSp	Parch	
0	1	0	3	Braund, Mr. Owen Harris	male	1	22.0	1	0	A/5
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	0	38.0	1	0	PC
2	3	1	3	Heikkinen, Miss. Laina	female	0	26.0	0	0	STC 31
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	0	35.0	1	0	1
4	5	0	3	Allen, Mr. William Henry	male	1	35.0	0	0	3
...	...	...	...	...	...	...	...	...	...	
886	887	0	2	Montvila, Rev. Juozas	male	1	27.0	0	0	2
887	888	1	1	Graham, Miss. Margaret Edith	female	0	19.0	0	0	1
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	0	29.0	1	2	
889	890	1	1	Behr, Mr. Karl Howell	male	1	26.0	0	0	1
890	891	0	3	Dooley, Mr. Patrick	male	1	32.0	0	0	3

891 rows × 15 columns

In [38]:

```
df1=pd.get_dummies(df,columns=['Embarked'], drop_first=True)
df1.head()
```

Out[38]:

	PassengerId	Survived	Pclass	Name	Sex	Gender_New	Age	SibSp	Parch	Ti
0	1	0	3	Braund, Mr. Owen Harris	male	1	22.0	1	0	A/5 21
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	0	38.0	1	0	PC 17
2	3	1	3	Heikkinen, Miss. Laina	female	0	26.0	0	0	STON/ 3101
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	0	35.0	1	0	113
4	5	0	3	Allen, Mr. William Henry	male	1	35.0	0	0	373

UNIVARIATE ANALYSIS

HOW MANY PEOPLE SURVIVED & HOW MANY DIED?

In [40]:

```
df['Survived'].value_counts()
```

Out[40]:

```
0    549
1    342
Name: Survived, dtype: int64
```

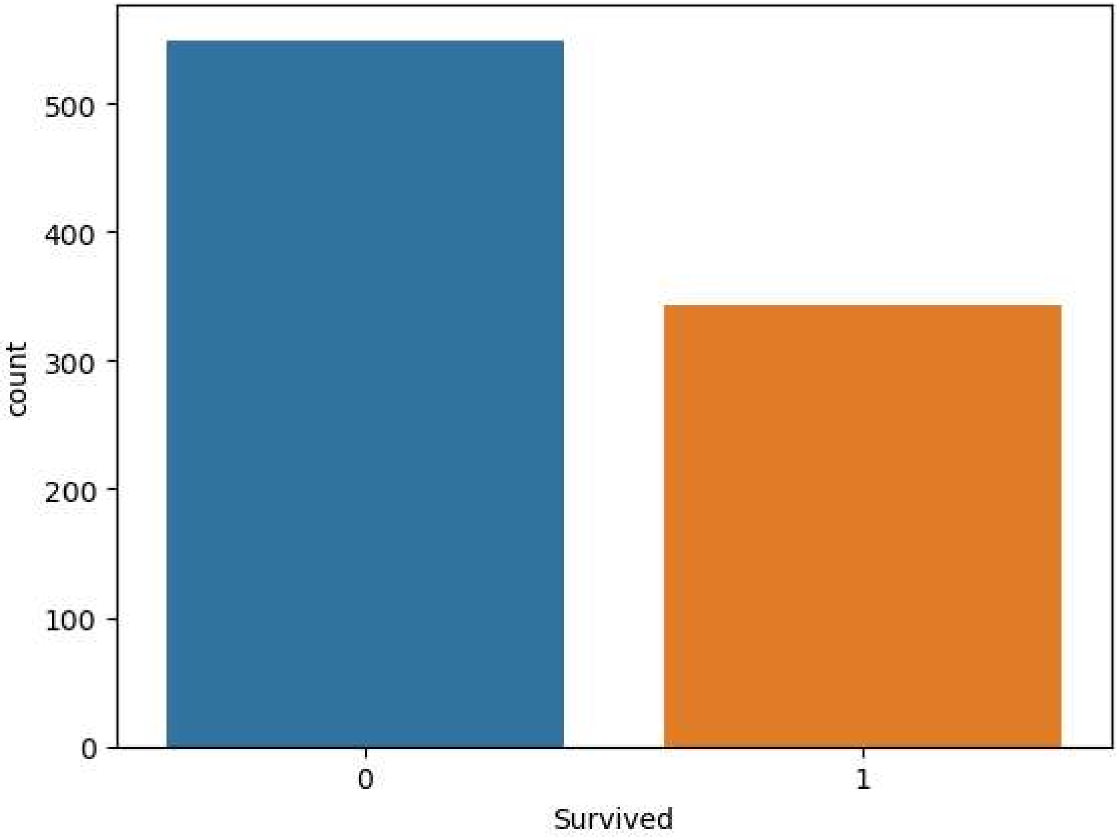
In [41]:

```
import matplotlib.pyplot as plt
```

In [42]:
```
sns.countplot(df['Survived']) #USE COUNTPLOT FOR CATEGORICAL VARIABLE
```

```
C:\Users\dell\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futur
eWarning: Pass the following variable as a keyword arg: x. From version 0.
12, the only valid positional argument will be `data`, and passing other a
rguments without an explicit keyword will result in an error or misinterpr
etation.
  warnings.warn(
```

Out[42]:
```
<AxesSubplot:xlabel='Survived', ylabel='count'>
```



HOW MANY PASSENGERS WERE IN FIRST CLASS, SECOND CLASS & THIRD CLASS?

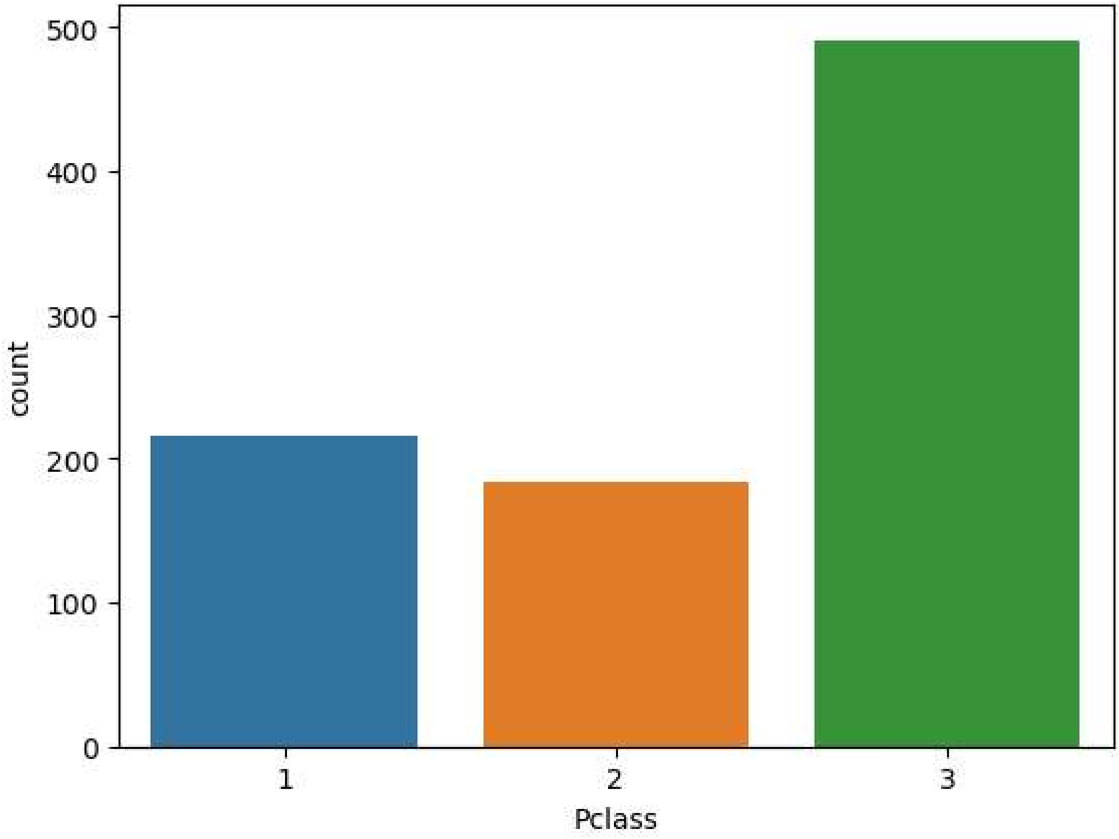
In [43]:
```
df['Pclass'].value_counts()
```

Out[43]:
```
3    491
1    216
2    184
Name: Pclass, dtype: int64
```

In [44]:
```
sns.countplot(df['Pclass'])
```

```
C:\Users\dell\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futur
eWarning: Pass the following variable as a keyword arg: x. From version 0.
12, the only valid positional argument will be `data`, and passing other a
rguments without an explicit keyword will result in an error or misinterpr
etation.
  warnings.warn(
```

Out[44]:
```
<AxesSubplot:xlabel='Pclass', ylabel='count'>
```



NO. OF MALE & FEMALE PASSENGERS

In [45]:
```
df['Sex'].value_counts()
```

Out[45]:
```
male      577
female    314
Name: Sex, dtype: int64
```

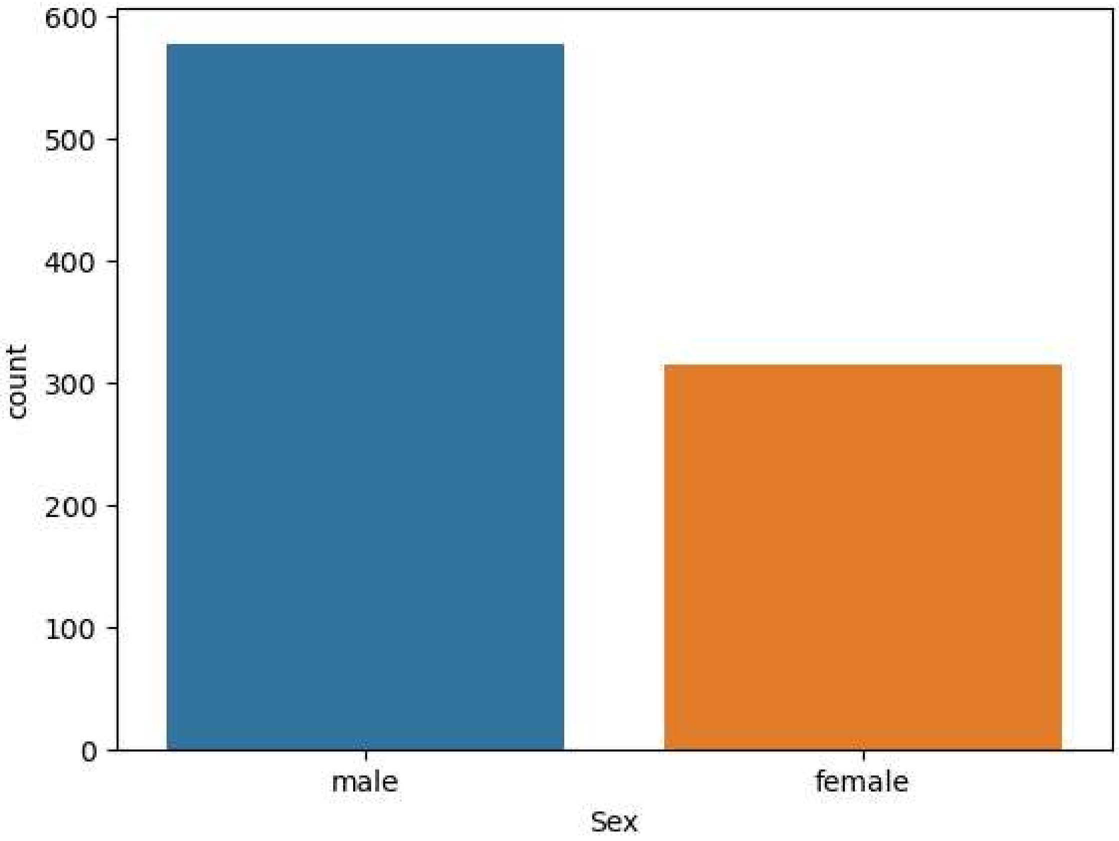
In [46]:

```
sns.countplot(df['Sex'])
```

```
C:\Users\dell\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futur
eWarning: Pass the following variable as a keyword arg: x. From version 0.
12, the only valid positional argument will be `data`, and passing other a
rguments without an explicit keyword will result in an error or misinterpr
etation.
  warnings.warn(
```

Out[46]:

```
<AxesSubplot:xlabel='Sex', ylabel='count'>
```

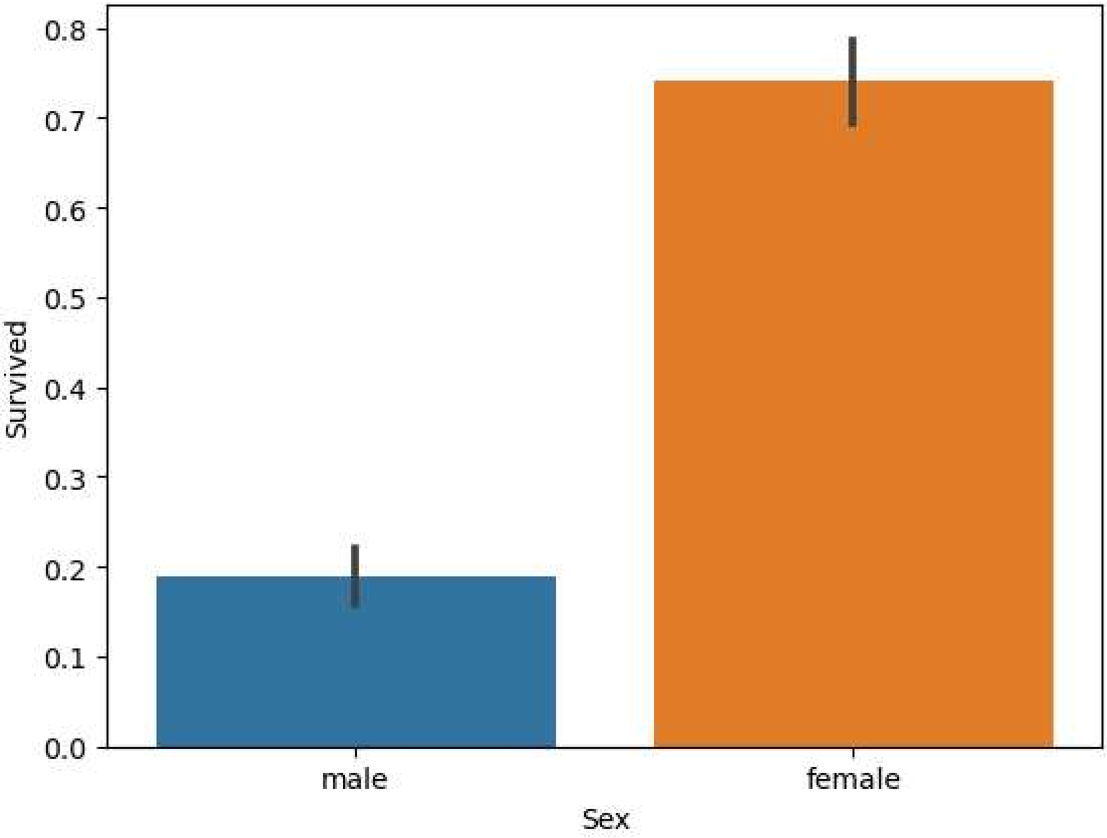


BIVARIATE ANALYSIS

WHO HAS BETTER CHANCE OF SURVIVAL. MALE OR FEMALE?

In [48]:
```
sns.barplot(y='Survived',x='Sex',data=df)
```

Out[48]:
```
<AxesSubplot:xlabel='Sex', ylabel='Survived'>
```



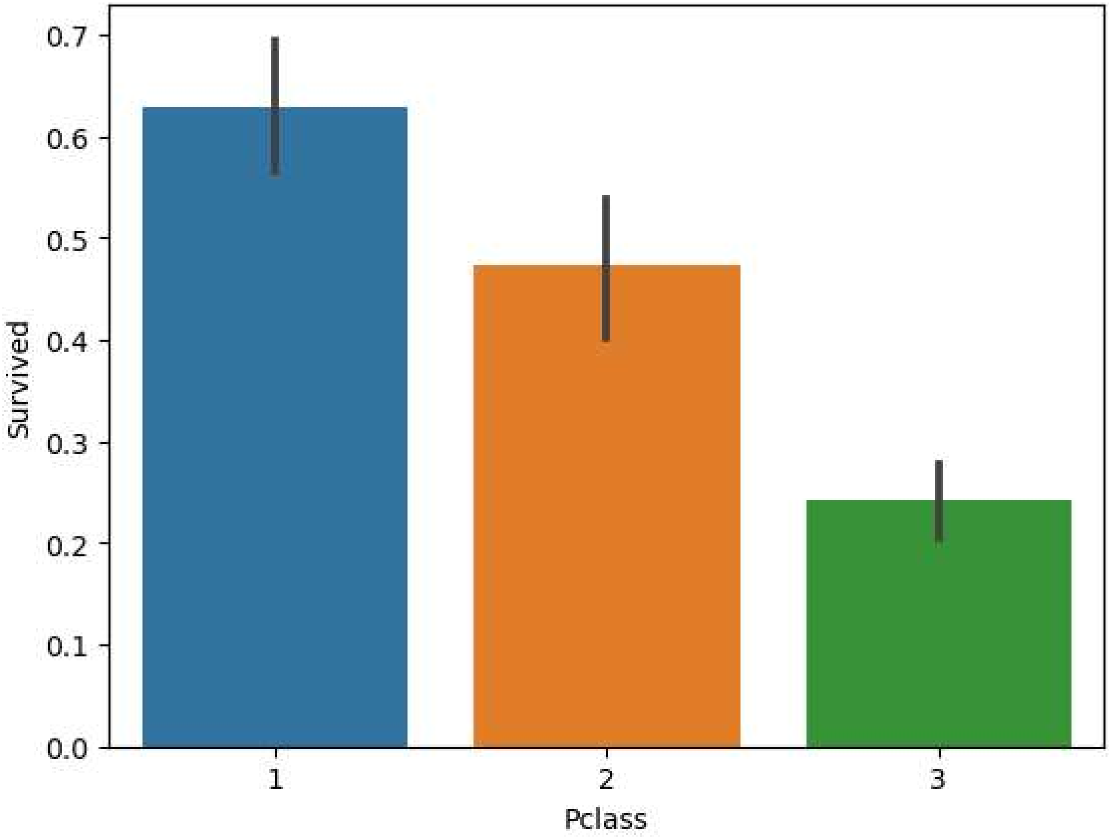
In [ ]:
```
WHICH PASSENGER CLASS HAS BETTER CHANCE OF SURVIVAL?
```

In [49]:

```
sns.barplot(y='Survived',x='Pclass',data=df)
```

Out[49]:

```
<AxesSubplot:xlabel='Pclass', ylabel='Survived'>
```



FEATURE ENGINEERING

In [50]:

```
df['Family_Size']=df['SibSp'] + df['Parch']
df.head()
```

Out[50]:

	PassengerId	Survived	Pclass	Name	Sex	Gender_New	Age	SibSp	Parch	Ti
0	1	0	3	Braund, Mr. Owen Harris	male	1	22.0	1	0	A/5 21
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	0	38.0	1	0	PC 17
2	3	1	3	Heikkinen, Miss. Laina	female	0	26.0	0	0	STON/ 3101
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	0	35.0	1	0	113
4	5	0	3	Allen, Mr. William Henry	male	1	35.0	0	0	373

In [51]:

```
#FARE PER PERSON
df['Fare_per_person']=df['Fare']/(df['Family_Size'] + 1)
df.head()
```

Out[51]:

	PassengerId	Survived	Pclass	Name	Sex	Gender_New	Age	SibSp	Parch	Ti
0	1	0	3	Braund, Mr. Owen Harris	male	1	22.0	1	0	A/5 21
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	0	38.0	1	0	PC 17
2	3	1	3	Heikkinen, Miss. Laina	female	0	26.0	0	0	STON/ 3101
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	0	35.0	1	0	113
4	5	0	3	Allen, Mr. William Henry	male	1	35.0	0	0	373

In [ ]: