In [3]:
```
import pandas as pd
z=pd.read_csv("C:\\Users\\ravit\\Downloads\\Titanic.csv")
display(z)
```

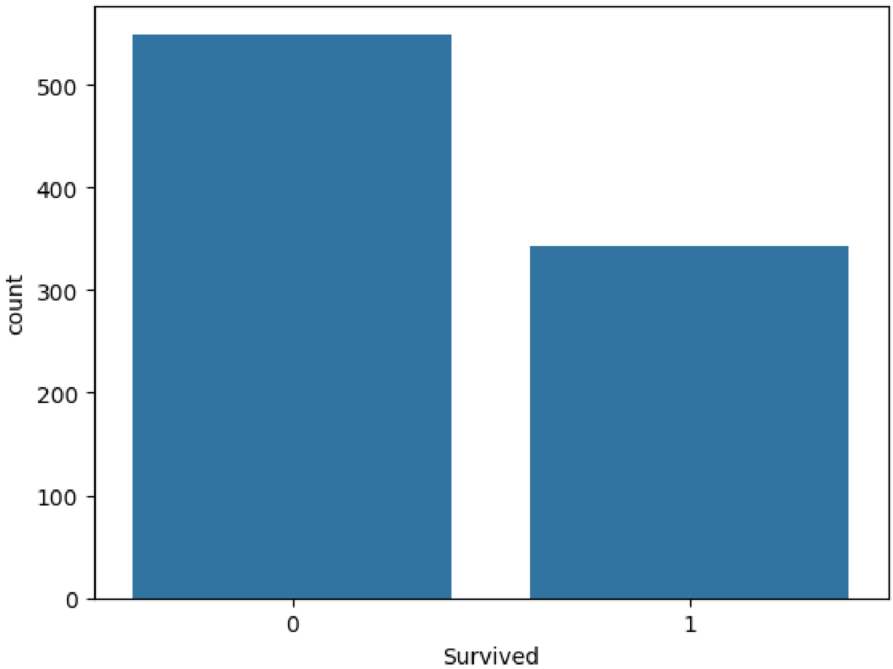
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

In [5]:
```
#displaying top 10 rows using head()
display(z.head(10))
```

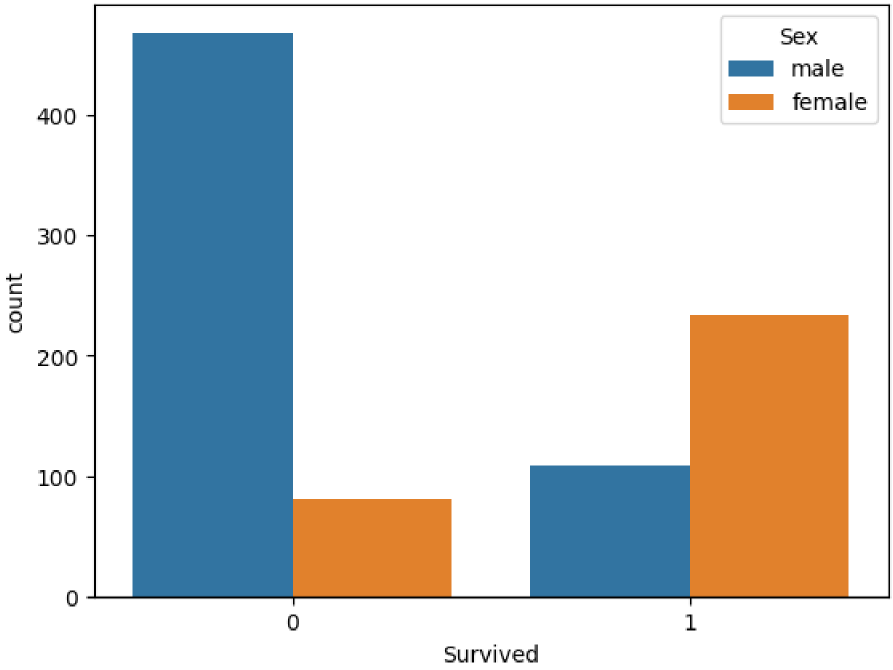
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN	S
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN	C

In [7]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.countplot(x='Survived',data=z)
plt.show()
```

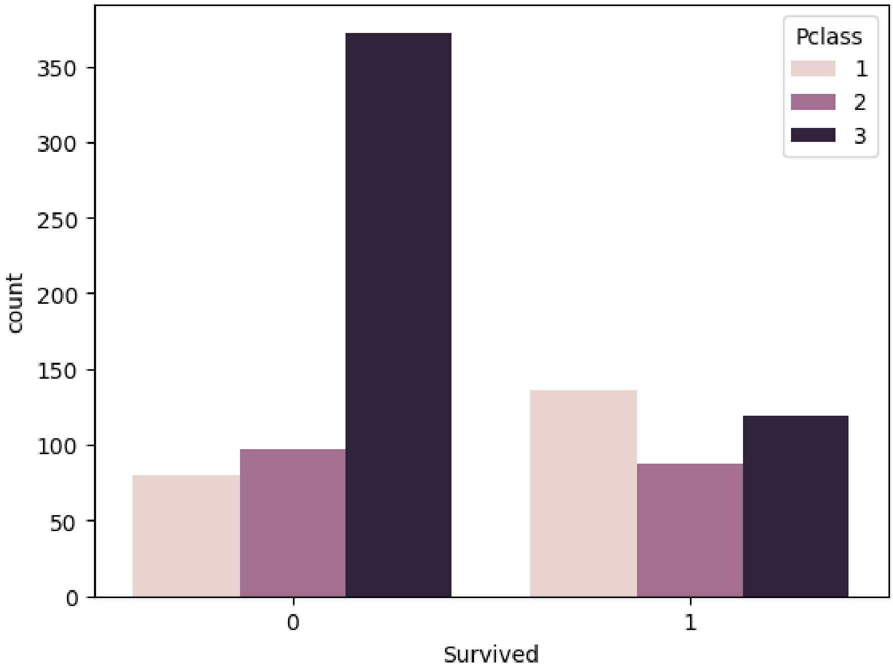


In [ ]:
```
#hear we can see unsurvived persons are more than survival persons
```
In [9]:
```
sns.countplot(x='Survived',hue='Sex',data=z)
```

Out[9]:
<Axes: xlabel='Survived', ylabel='count'>

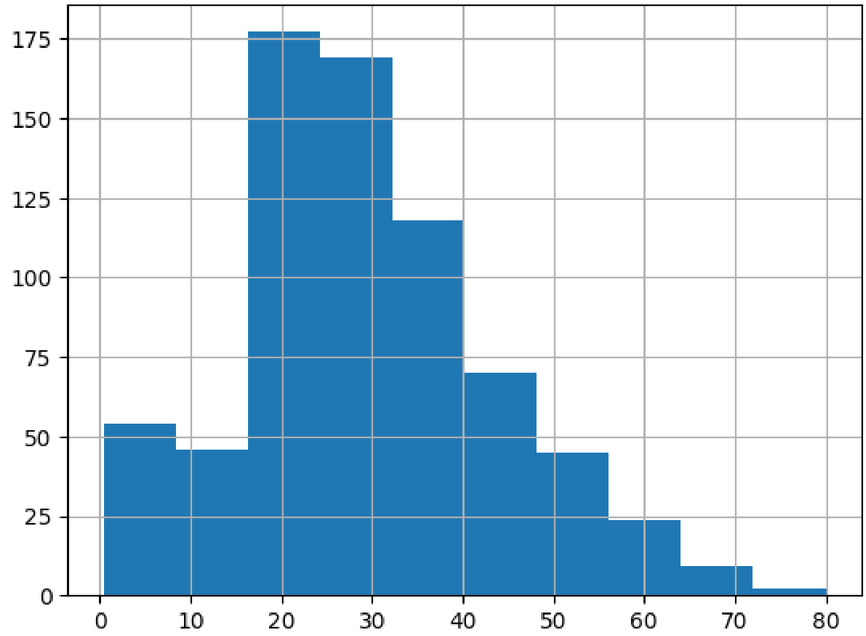


In [10]:
```
sns.countplot(x='Survived',hue='Pclass',data=z)
```
Out[10]:
<Axes: xlabel='Survived', ylabel='count'>

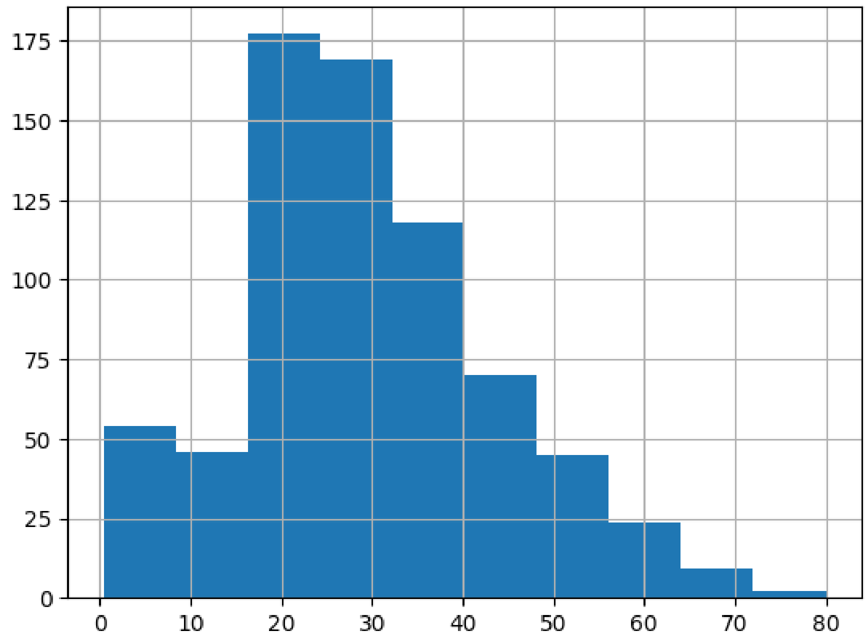


In [11]:
#here we can see mostly first class people are survived than the 3rd class people.
In [15]:
```
z['Age'].hist()
```

Out[15]:
<Axes: >

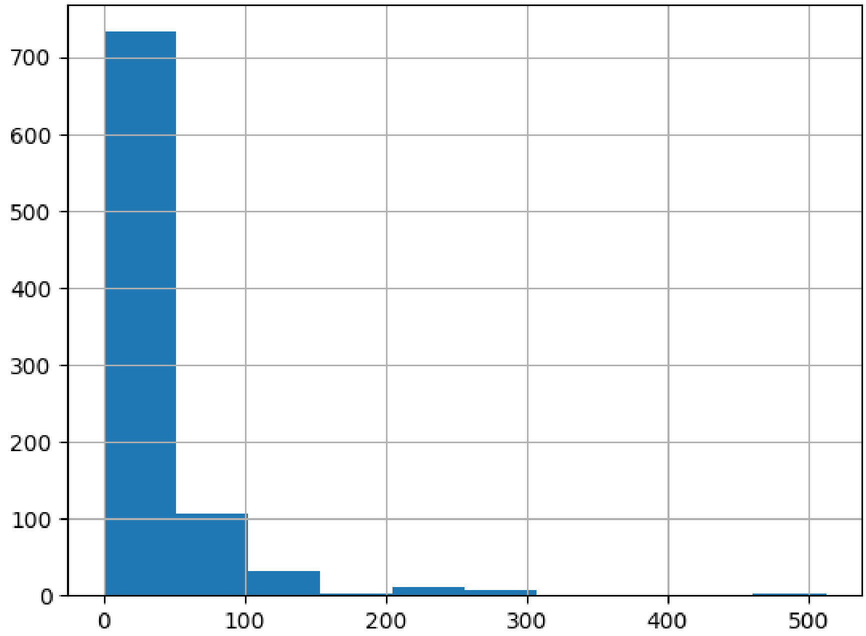


In [15]:
```
z['Age'].hist()
```
Out[15]:
<Axes: >

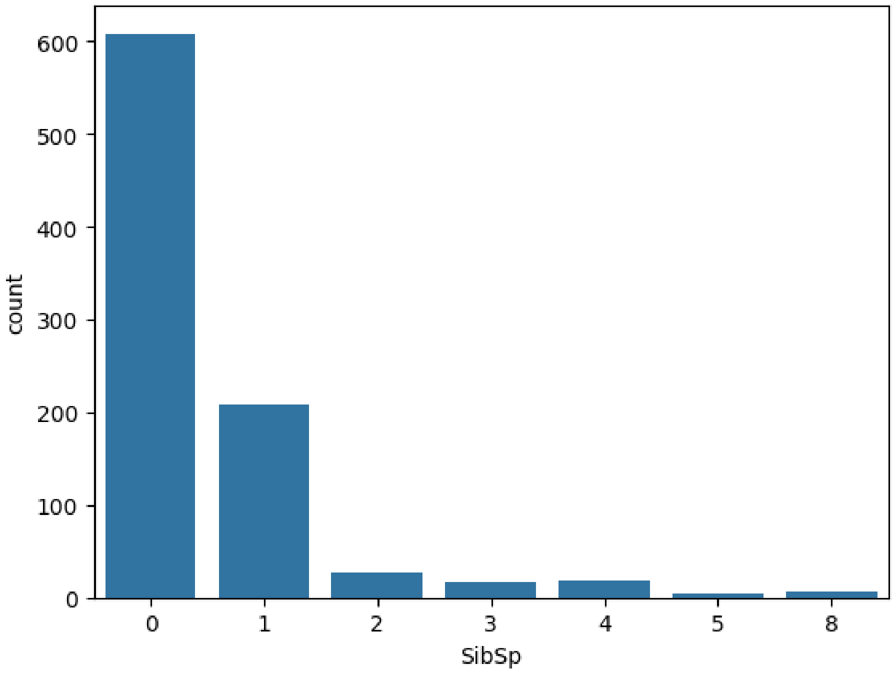


In [16]:
```
#we can observe mostly 20-40 age group passengers are more
#very few passengers are prsent in the age gap 70-80
```
In [17]:
```
z['Fare'].hist()
```

Out[17]:
<Axes: >

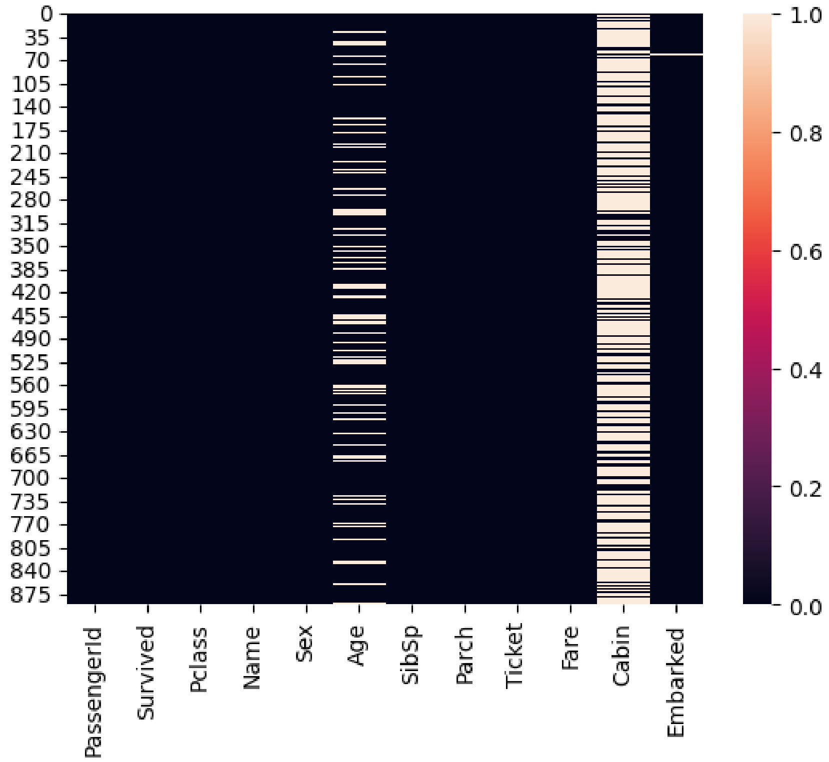


In [18]:
#we can observe most of the tickets are brought under fare
In [21]:
```
sns.countplot(x='SibSp',data=z)
```
Out[21]:
<Axes: xlabel='SibSp', ylabel='count'>

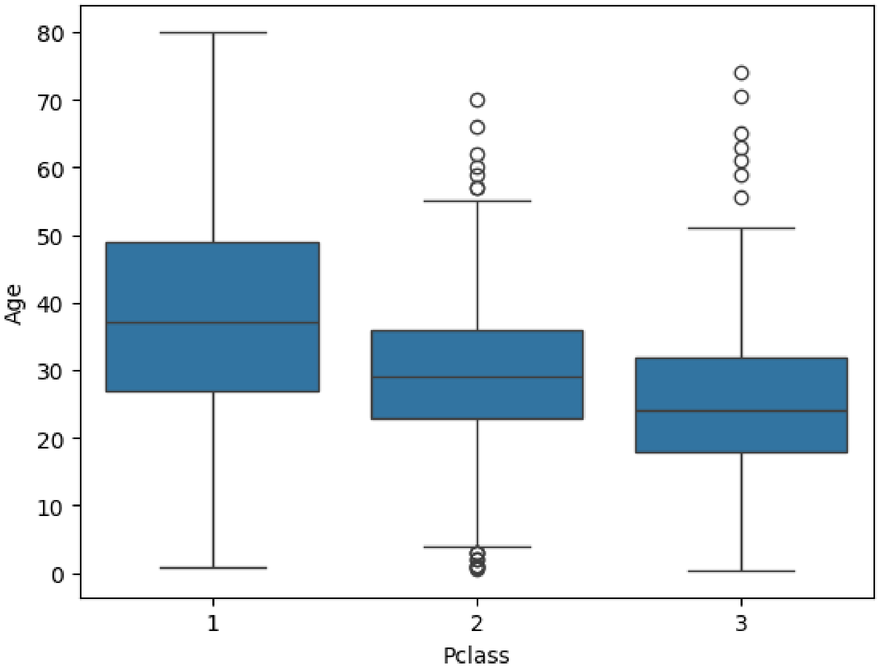


In [22]:
 # we observe that most of the passengers do not have siblings abroad.
In [ ]:
#performing data wrangling
In [24]:
```
z.isnull().sum()
```
Out[24]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```
In [27]:
```
sns.heatmap(z.isnull())
```

Out[27]:
<Axes: >

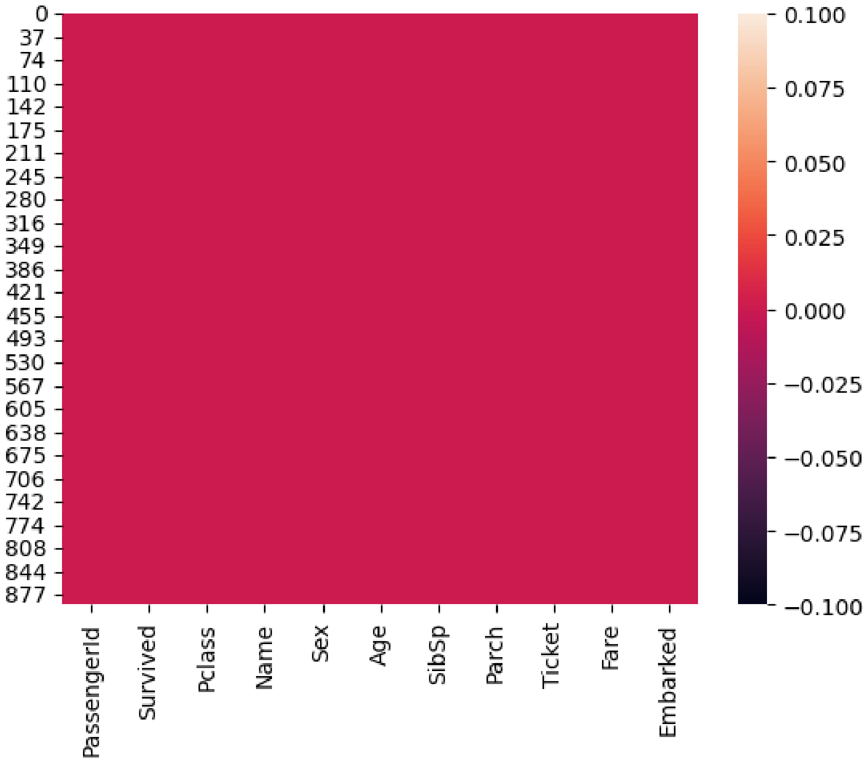


In [28]:
#here light yellow color shows that null values are more in cabin column than the age column
In [29]:
```
sns.boxplot(x='Pclass',y='Age',data=z)
```
Out[29]:
<Axes: xlabel='Pclass', ylabel='Age'>



In [30]:
#here we observe older age group are travelling more in class 1 and 2.
In [43]:
```
z.dropna(inplace=True)
```
In [45]:
```
sns.heatmap(z.isnull(),cmap='rocket')
```

Out[45]:
<Axes: >



In [46]:
```
z.isnull().sum()
```
Out[46]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```
In [ ]:
```
#ONE HOT ENCODING
#One-Hot Encoding simply creates one column for every possible value and put a 1 in the appropriate column.
#I will convert a few columns into categorical data to perform Logistic Regression, as Logistic Regression takes categorical /binary values.
#get_dummies() function is used to Convert categorical variable into dummy/indicator variables.
```
In [153]:
```
pd.get_dummies(z['Sex'],dtype=int).head()
```
Out[153]:

	female	male
0	0	1
1	1	0
2	1	0
3	1	0
4	0	1

In [157]:
```
pd.get_dummies(z['Embarked'],dtype=int).head()
emb=pd.get_dummies(z['Embarked'],dtype=int,drop_first=True).head()
emb
```

Out[157]:

	Q	S
0	0	1
1	0	0
2	0	1
3	0	1
4	0	1

In [158]:
```
pd.get_dummies(z['Pclass'],dtype=int).head()
pcl=pd.get_dummies(z['Pclass'],dtype=int,drop_first=True).head()
pcl
```
Out[158]:

	2	3
0	0	1
1	0	0
2	0	1
3	0	0
4	0	1

In [199]:
```
conct=pd.concat([z,sex,emb,pcl],axis=1)
```
In [200]:
```
conct.head()
```
Out[200]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	...	male	2	3
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	...	True	False	True
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	...	False	False	False
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	...	False	False	True
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	...	False	False	False
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	...	True	False	True

5 rows × 21 columns

In [203]:
```
conct1=conct.drop(['Name','PassengerId','Pclass','Ticket','Sex','Embarked'],axis=1)
conct1
```

Out[203]:

	Survived	Age	SibSp	Parch	Fare	male	2	3	Q	S	male	Q	S	2	3
0	0	22.0	1	0	7.2500	True	False	True	False	True	1.0	0.0	1.0	0.0	1.0
1	1	38.0	1	0	71.2833	False	False	False	False	False	0.0	0.0	0.0	0.0	0.0
2	1	26.0	0	0	7.9250	False	False	True	False	True	0.0	0.0	1.0	0.0	1.0
3	1	35.0	1	0	53.1000	False	False	False	False	True	0.0	0.0	1.0	0.0	0.0
4	0	35.0	0	0	8.0500	True	False	True	False	True	1.0	0.0	1.0	0.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
885	0	39.0	0	5	29.1250	False	False	True	True	False	NaN	NaN	NaN	NaN	NaN
886	0	27.0	0	0	13.0000	True	True	False	False	True	NaN	NaN	NaN	NaN	NaN
887	1	19.0	0	0	30.0000	False	False	False	False	True	NaN	NaN	NaN	NaN	NaN
889	1	26.0	0	0	30.0000	True	False	False	False	False	NaN	NaN	NaN	NaN	NaN
890	0	32.0	0	0	7.7500	True	False	True	True	False	NaN	NaN	NaN	NaN	NaN

712 rows × 15 columns

In [204]:

```
conct1.drop(conct1.iloc[:,5:10],axis=1)
```

Out[204]:

	Survived	Age	SibSp	Parch	Fare
0	0	22.0	1	0	7.2500
1	1	38.0	1	0	71.2833
2	1	26.0	0	0	7.9250
3	1	35.0	1	0	53.1000
4	0	35.0	0	0	8.0500
...	...	...	...	...	...
885	0	39.0	0	5	29.1250
886	0	27.0	0	0	13.0000
887	1	19.0	0	0	30.0000
889	1	26.0	0	0	30.0000
890	0	32.0	0	0	7.7500

712 rows × 5 columns

In [3]:

```
import pandas as pd
z=pd.read_csv("C:\\Users\\ravit\\Downloads\\Titanic.csv")
sex=pd.get_dummies(z['Sex'],dtype=int).head()
display(sex)
pd.get_dummies(z['Embarked'],dtype=int).head()
emb=pd.get_dummies(z['Embarked'],dtype=int,drop_first=True).head()
display(emb)
pd.get_dummies(z['Pclass'],dtype=int).head()
pcl=pd.get_dummies(z['Pclass'],dtype=int,drop_first=True).head()
display(pcl)
conct=pd.concat([z,sex,emb,pcl],axis=1)
conct1=conct.drop(['Name','PassengerId','Pclass','Ticket','Sex','Embarked'],axis=1)
conct1
X=conct1.drop('Survived',axis=1)
y=conct1['Survived']
```

	female	male
0	0	1
1	1	0
2	1	0
3	1	0
4	0	1

	Q	S
0	0	1
1	0	0
2	0	1
3	0	1
4	0	1

	2	3
0	0	1
1	0	0
2	0	1
3	0	0
4	0	1

In [4]:
```
from sklearn.model_selection import train_test_split
```
In [5]:
```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.33,random_state=4)
```
In [6]:
```
from sklearn.linear_model import LogisticRegression
lm=LogisticRegression()
```
In [ ]:
```
#by performing machine learning algorithm we got 80% accuracy
```
In [10]:
```
# I conclude we got 80% accuracy total which makes our model good model to predict the values accurately
```
In [11]:
```
#Through visualization we found out that females have more chances of survrival than males,class1 have more chances of survival
```
In [12]:
```
#further other machine learning algorithms can be applied on same data set,emsemble algorithms to boost the performance of model
#and get good predictions
```
In [ ]: