In [72]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
titanic=sns.load_dataset('titanic')
titanic.head()
```

Out[72]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

In [73]:

```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
survived       891 non-null int64
pclass         891 non-null int64
sex            891 non-null object
age            714 non-null float64
sibsp          891 non-null int64
parch          891 non-null int64
fare           891 non-null float64
embarked       889 non-null object
class          891 non-null category
who            891 non-null object
adult_male     891 non-null bool
deck           203 non-null category
embark_town    889 non-null object
alive          891 non-null object
alone          891 non-null bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.6+ KB
```

In [74]:

```
titanic.describe()
```

Out[74]:

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

In [75]:

```
cat=titanic.select_dtypes(include=['object']).copy()
cat.head()
```

Out[75]:

	sex	embarked	who	embark_town	alive
0	male	S	man	Southampton	no
1	female	C	woman	Cherbourg	yes
2	female	S	woman	Southampton	yes
3	female	S	woman	Southampton	yes
4	male	S	man	Southampton	no

In [76]:

```
titanic.isnull()
```

Out[76]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False
887	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
888	False	False	False	True	False	False	False	False	False	False	False	True	False	False	False
889	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
890	False	False	False	False	False	False	False	False	False	False	False	True	False	False	False

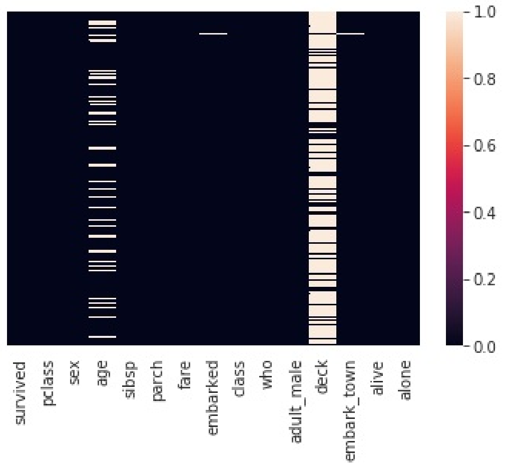
891 rows × 15 columns

In [77]:

```
sns.heatmap(titanic.isnull(),yticklabels=False)
```

Out[77]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23f166e50>
```



In [78]:

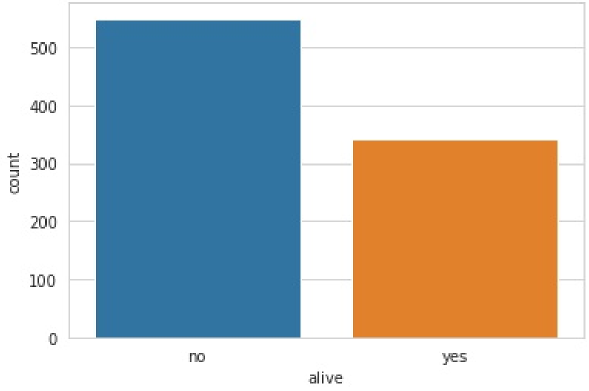
```
sns.set_style('whitegrid')
```

In [79]:

```
sns.countplot(x='alive',data=titanic)
```

Out[79]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23f086890>
```

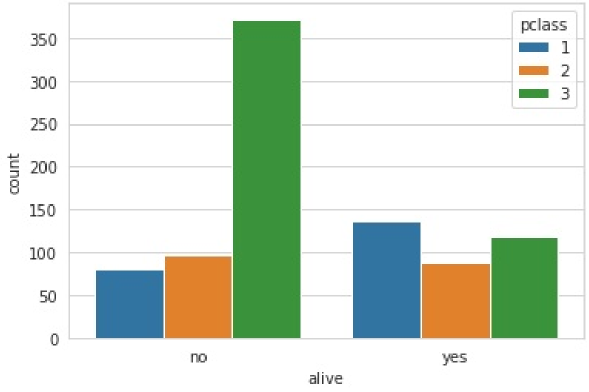


In [80]:

```
sns.countplot(x='alive',data=titanic,hue='pclass')
```

Out[80]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23f061e10>
```

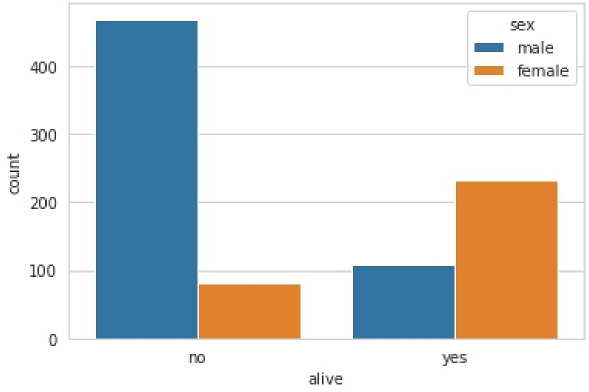


In [81]:

```
sns.countplot(x='alive',data=titanic,hue='sex')
```

Out[81]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23efc2510>
```

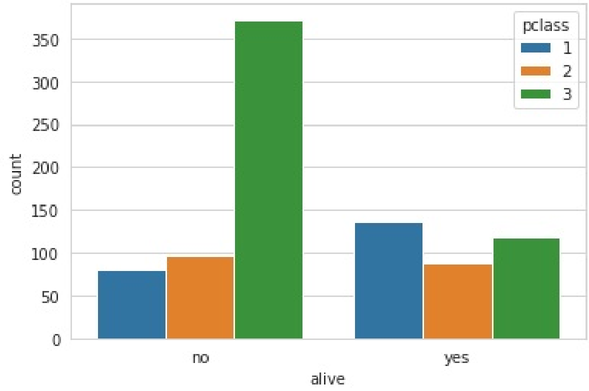


In [82]:

```
sns.countplot(x='alive',data=titanic,hue='pclass')
```

Out[82]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23efae210>
```

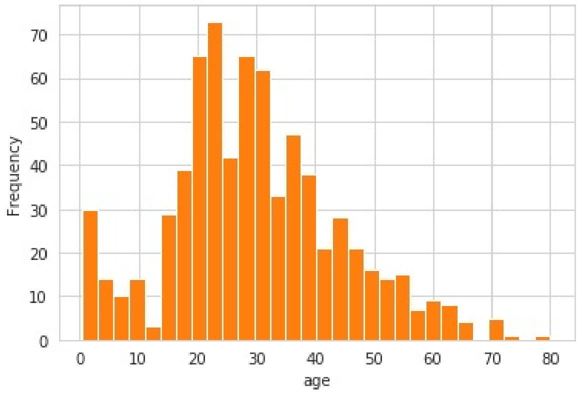


In [83]:

```
sns.distplot(titanic['age'].dropna(),kde=False,bins=30)
titanic['age'].plot.hist(bins=30)
```

Out[83]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23ef00c10>
```

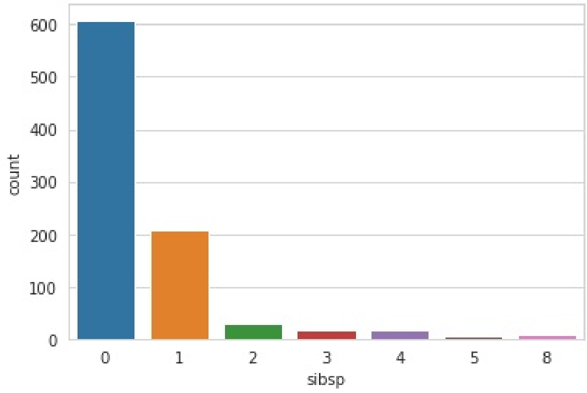


In [84]:

```
sns.countplot(x='sibsp',data=titanic)
```

Out[84]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23ef56390>
```

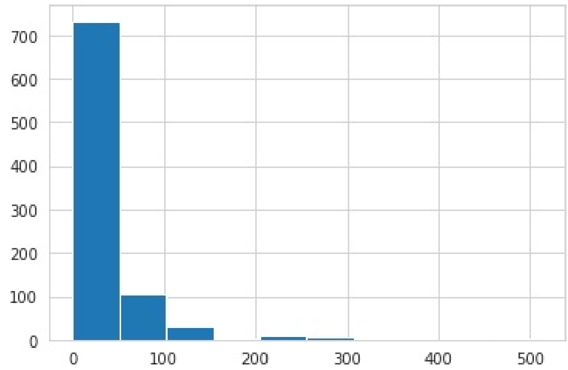


In [85]:

```
titanic['fare'].hist()
```

Out[85]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23ef09990>
```

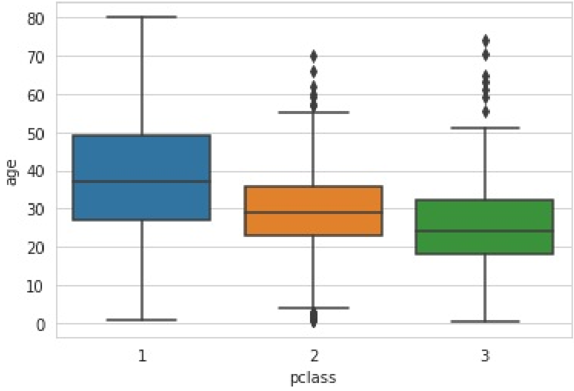


In [86]:

```
sns.boxplot(x='pclass',y='age',data=titanic)
```

Out[86]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa2432a17d0>
```



In [87]:

```
#fill missing age values by its mean values
```

In [88]:

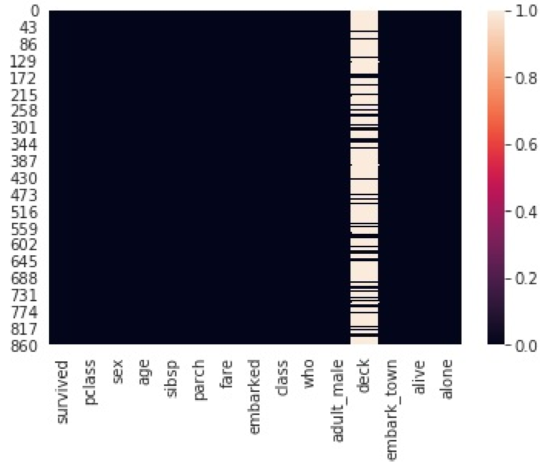
```
def impute_age(cols):
    age=cols[0]
    pclass=cols[1]
    if pd.isnull(age):
        if pclass==1:
            return 37
        elif pclass==2:
            return 29
        else:
            return 24
    else:
        return age
```

In [89]:

```
titanic['age']=titanic[['age','pclass']].apply(impute_age,axis=1)#calls function giving age and pclass as arguments
sns.heatmap(titanic.isnull())
```

Out[89]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23ecad0d0>
```



In [90]:

```
titanic.drop('deck',axis=1,inplace=True)
```

In [91]:

```
titanic.head()
```

Out[91]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	Southampton	no	True

In [92]:

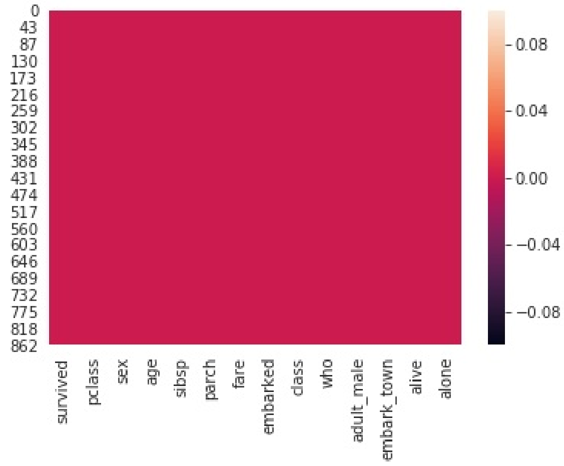
```
titanic.dropna(inplace=True)
```

In [93]:

```
sns.heatmap(titanic.isnull())
```

Out[93]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa23eb81450>
```



In [94]:

```
sex=pd.get_dummies(titanic['sex'],drop_first=True)
```

In [95]:

```
embark_town=pd.get_dummies(titanic['embark_town'],drop_first=True)
```

In [96]:

```
titanic.head()
```

Out[96]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	Southampton	no	True

In [97]:

```
titanic=pd.concat([titanic,sex,embark_town],axis=1)
```

In [98]:

```
alone=pd.get_dummies(titanic['alone'],drop_first=True)
```

In [99]:

```
titanic.head()
```

Out[99]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	embark_town	alive	alone	male	Quee
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	Southampton	no	False	1	
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False	0	
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	Southampton	yes	True	0	
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	Southampton	yes	False	0	
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	Southampton	no	True	1	

In [100]:

```
titanic.drop(['sex','embark_town','alone'],axis=1,inplace=True)
```

In [101]:

```
class1=pd.get_dummies(titanic['class'],drop_first=True)
```

In [102]:

```
titanic.drop(['class'],axis=1,inplace=True)
```

In [103]:

```
titanic.head()
```

Out[103]:

	survived	pclass	age	sibsp	parch	fare	embarked	who	adult_male	alive	male	Queenstown	Southampton
0	0	3	22.0	1	0	7.2500	S	man	True	no	1	0	1
1	1	1	38.0	1	0	71.2833	C	woman	False	yes	0	0	0
2	1	3	26.0	0	0	7.9250	S	woman	False	yes	0	0	1
3	1	1	35.0	1	0	53.1000	S	woman	False	yes	0	0	1
4	0	3	35.0	0	0	8.0500	S	man	True	no	1	0	1

In [104]:

```
who=pd.get_dummies(titanic['who'],drop_first=True)
```

In [105]:

```
titanic=pd.concat([titanic,who,class1],axis=1)
```

In [108]:

```
titanic.drop(['who'],axis=1,inplace=True)
```

In [110]:

```
titanic.drop(['adult_male'],axis=1,inplace=True)
```

In [112]:

```
embarked=pd.get_dummies(titanic['embarked'],drop_first=True)
```

In [113]:

```
titanic=pd.concat([titanic,embarked],axis=1)
```

In [115]:

```
titanic.drop(['embarked'],axis=1,inplace=True)
```

In [116]:

```
titanic.head()
```

Out[116]:

	survived	pclass	age	sibsp	parch	fare	alive	male	Queenstown	Southampton	man	woman	Second	Third	Q	S
0	0	3	22.0	1	0	7.2500	no	1	0	1	1	0	0	1	0	1
1	1	1	38.0	1	0	71.2833	yes	0	0	0	0	1	0	0	0	0
2	1	3	26.0	0	0	7.9250	yes	0	0	1	0	1	0	1	0	1
3	1	1	35.0	1	0	53.1000	yes	0	0	1	0	1	0	0	0	1
4	0	3	35.0	0	0	8.0500	no	1	0	1	1	0	0	1	0	1

In [117]:

```
titanic.drop('alive',axis=1)
```

Out[117]:

	survived	pclass	age	sibsp	parch	fare	male	Queenstown	Southampton	man	woman	Second	Third	Q	S
0	0	3	22.0	1	0	7.2500	1	0	1	1	0	0	1	0	1
1	1	1	38.0	1	0	71.2833	0	0	0	0	1	0	0	0	0
2	1	3	26.0	0	0	7.9250	0	0	1	0	1	0	1	0	1
3	1	1	35.0	1	0	53.1000	0	0	1	0	1	0	0	0	1
4	0	3	35.0	0	0	8.0500	1	0	1	1	0	0	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	27.0	0	0	13.0000	1	0	1	1	0	1	0	0	1
887	1	1	19.0	0	0	30.0000	0	0	1	0	1	0	0	0	1
888	0	3	24.0	1	2	23.4500	0	0	1	0	1	0	1	0	1
889	1	1	26.0	0	0	30.0000	1	0	0	1	0	0	0	0	0
890	0	3	32.0	0	0	7.7500	1	1	0	1	0	0	1	1	0

889 rows × 15 columns

In [118]:

```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 889 entries, 0 to 890
Data columns (total 16 columns):
survived       889 non-null int64
pclass         889 non-null int64
age            889 non-null float64
sibsp          889 non-null int64
parch          889 non-null int64
fare           889 non-null float64
alive          889 non-null object
male           889 non-null uint8
Queenstown     889 non-null uint8
Southampton    889 non-null uint8
man            889 non-null uint8
woman          889 non-null uint8
Second         889 non-null uint8
Third          889 non-null uint8
Q              889 non-null uint8
S              889 non-null uint8
dtypes: float64(2), int64(4), object(1), uint8(9)
memory usage: 63.4+ KB
```

In [119]:

```
X=titanic[[
'pclass','age','sibsp','parch','fare','male','Queenstown','Southampton','man','woman','Second','Third','Q','S']]
y=titanic['survived']
```

In [178]:

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_state=101)
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
lr.fit(X_train,y_train)
predictions=lr.predict(X_test)
from sklearn.metrics import classification_report
print(classification_report(y_test,predictions))
```

```
              precision    recall  f1-score   support

           0       0.83      0.93      0.87       163
           1       0.86      0.69      0.77       104

    accuracy                           0.84       267
   macro avg       0.84      0.81      0.82       267
weighted avg       0.84      0.84      0.83       267


/home/anudeep/anaconda3/lib/python3.7/site-packages/sklearn/linear_model/logistic.py:432: FutureWarn
ing: Default solver will be changed to 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)
```

In [179]:

```
from sklearn.metrics import confusion_matrix
```

In [180]:

```
ccc
```

```
[[151  12]
 [ 32  72]]
```

In [181]:

```
print(((151+72)/(151+12+72+32))*100) #83.5% accuracy
```

```
83.52059925093633
```

In [182]:

```
score=lr.score(X_test,y_test)
print(score)
```

```
0.8352059925093633
```

In [ ]:

```
```