In [1]:
```
1.import the necessary libraries

2.import the dataset

3.Handling null values

4.outlier detection---surya

5.Seperate Dependent and independent variables

6.Encoding

7.splitting into training and testing set

8.Feature scaling
```

```
  Cell In[1], line 1
    1.import the necessary libraries
     ^
SyntaxError: invalid decimal literal
```

1.import the necessary libraries

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

2.import the dataset

In [4]:
```
dataset=pd.read_csv("Titanic-Dataset.csv")
```

In [5]:
```
dataset
```

Out[5]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fa
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25(
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.28:
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.10(
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.05(
...	...	...	...	...	...	...	...	...	...	
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00(
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00(
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45(
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00(
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75(

891 rows × 12 columns

In [6]:
```
dataset.head()
```

Out[6]:

	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
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500

In [ ]:
```
```

In [7]:
```
dataset.tail()
```

Out[7]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Ca
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	I
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	I
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	I

In [8]:
```
dataset.shape
```

Out[8]:
```
(891, 12)
```

3.Handling null values

In [9]:
```
dataset.isnull().any()
```

Out[9]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age             True
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin           True
Embarked        True
dtype: bool
```

In [10]:
```
dataset.isnull().sum()
```

Out[10]:
```
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 [11]:
```
dataset["Age"].fillna(dataset["Age"].mean(),inplace=True)
```

In [12]:
```
dataset["Age"]
```

Out[12]:
```
0      22.000000
1      38.000000
2      26.000000
3      35.000000
4      35.000000
         ...
886    27.000000
887    19.000000
888    29.699118
889    26.000000
890    32.000000
Name: Age, Length: 891, dtype: float64
```

In [13]:
```
dataset.isnull().sum()
```

Out[13]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age              0
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

In [14]:
```
dataset["Cabin"].fillna(dataset["Cabin"].mode()[0],inplace=True)
```

In [15]:
```
dataset.isnull().any()
```

Out[15]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age            False
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin          False
Embarked        True
dtype: bool
```

In [16]:
```
dataset.isnull().sum()
```

Out[16]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       2
dtype: int64
```

In [17]:
```
dataset["Embarked"].fillna(dataset["Embarked"].mode()[0],inplace=True)
```

In [18]:
```
dataset.drop(['Cabin'],axis=1,inplace=True)
```

In [19]:
```
dataset.isnull().any()
```

Out[19]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age            False
SibSp          False
Parch          False
Ticket         False
Fare           False
Embarked       False
dtype: bool
```

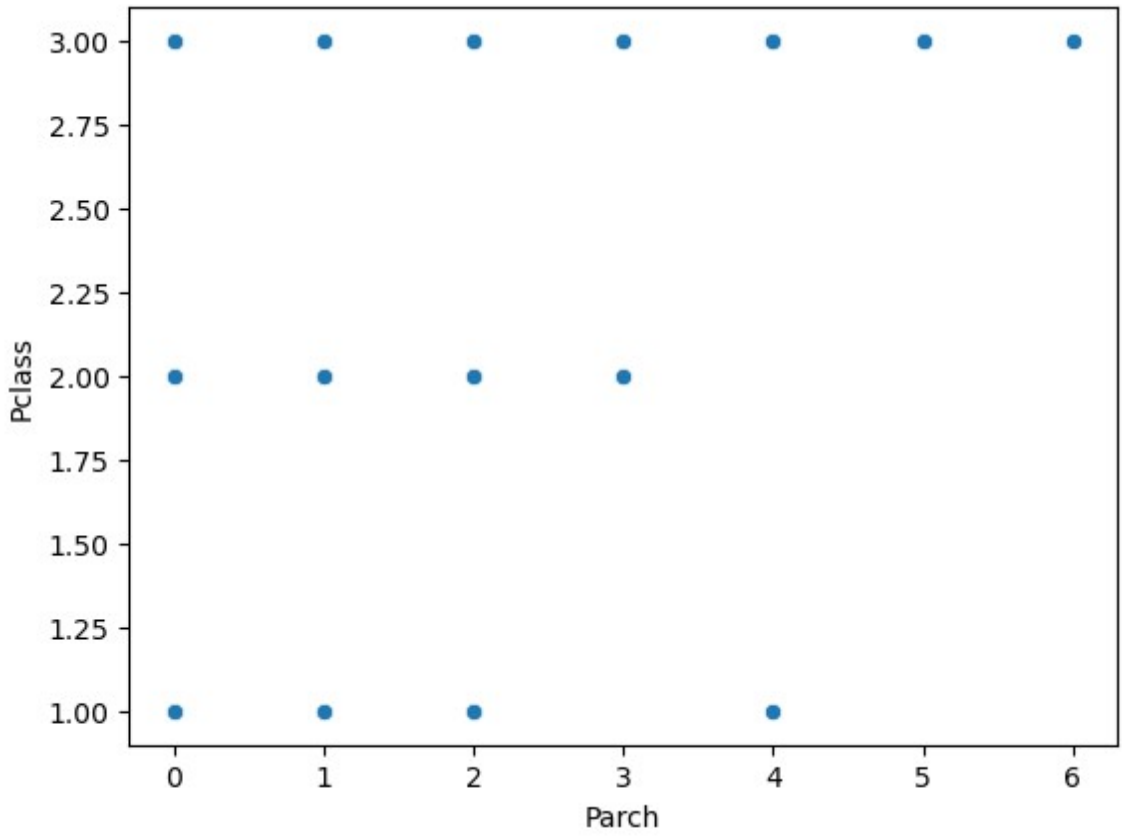
4 Data visulaization

In [20]:
```
corr=dataset.corr()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\3512126831.py:1: FutureWarni
ng: The default value of numeric_only in DataFrame.corr is deprecated. In
a future version, it will default to False. Select only valid columns or s
pecify the value of numeric_only to silence this warning.
  corr=dataset.corr()
```

In [21]:
```
sns.scatterplot(x="Parch",y="Pclass",data=dataset)
```

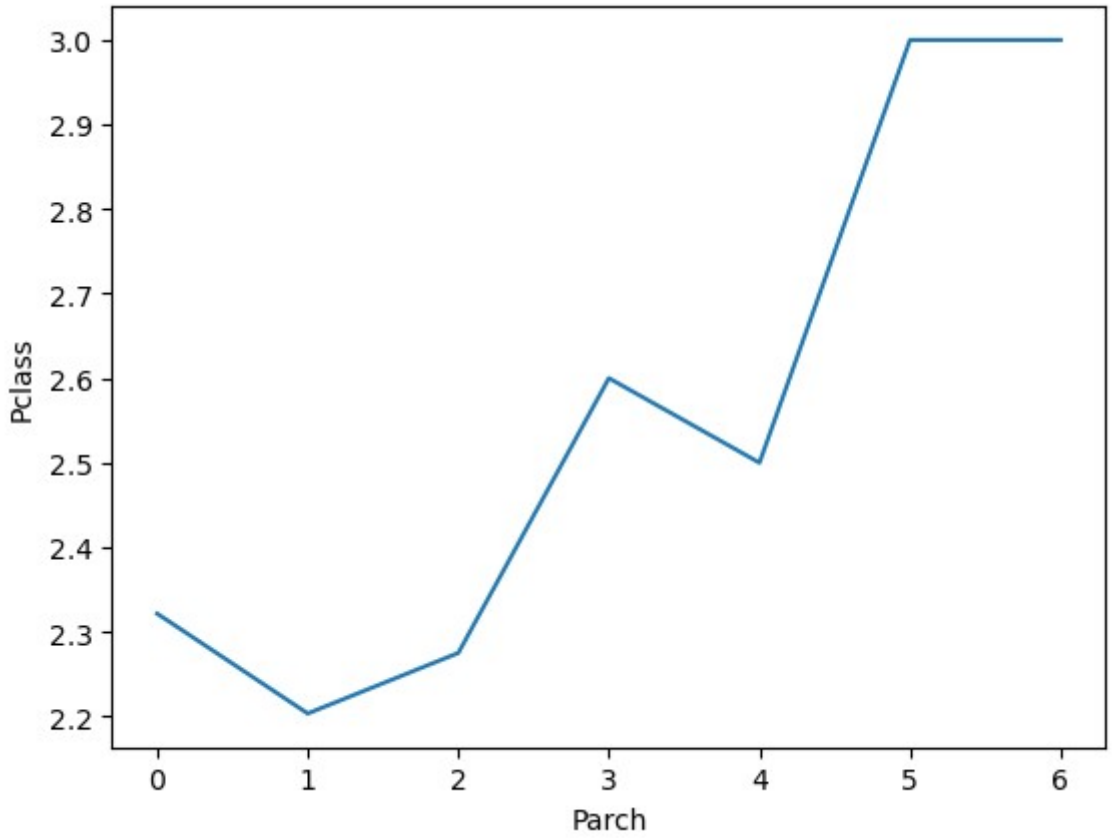
Out[21]:
```
<Axes: xlabel='Parch', ylabel='Pclass'>
```



In [22]:
```
sns.lineplot(x="Parch",y="Pclass",data=dataset,ci=None)
```

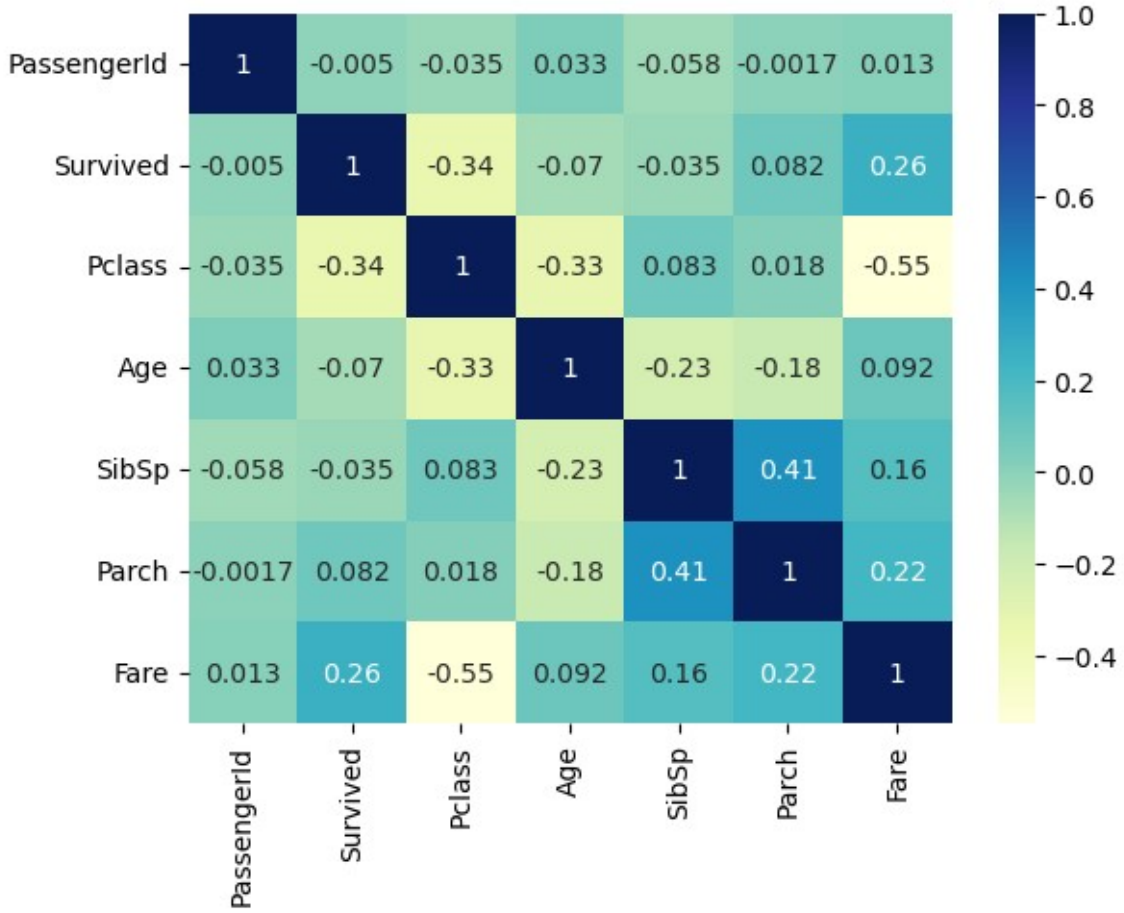
```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\1346139417.py:1: FutureWarni
ng:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.lineplot(x="Parch",y="Pclass",data=dataset,ci=None)
```

Out[22]:
```
<Axes: xlabel='Parch', ylabel='Pclass'>
```



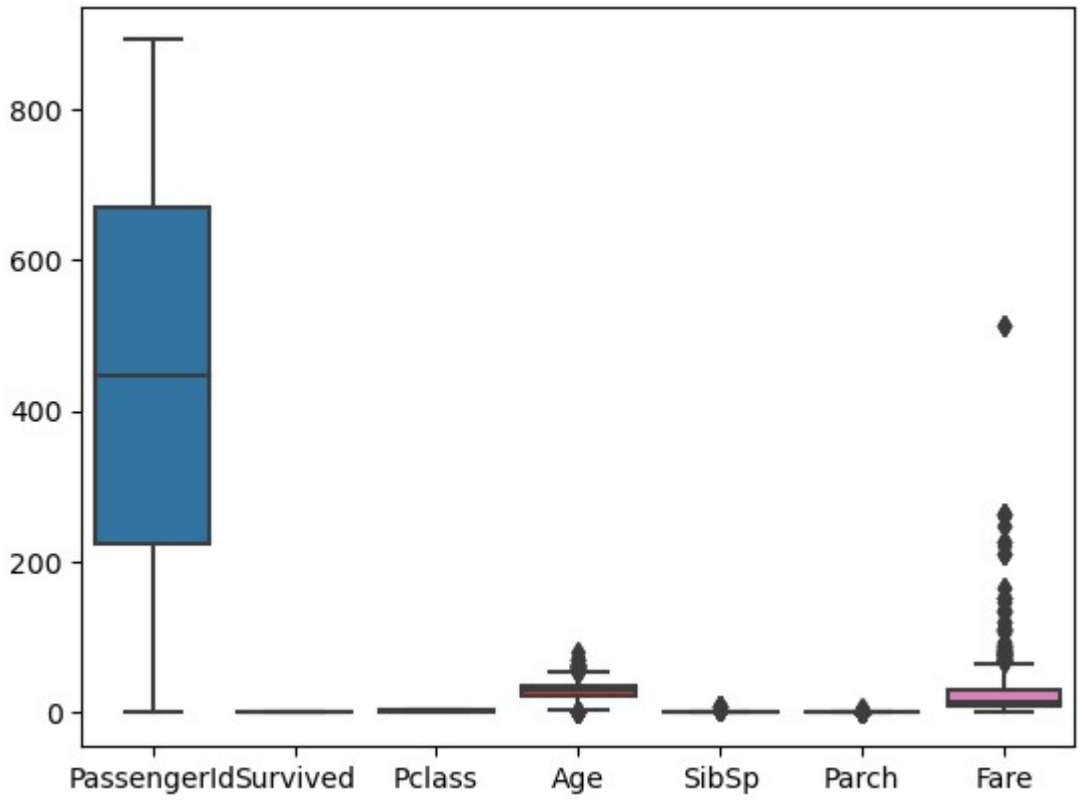
In [23]:
```
sns.heatmap(corr,annot=True,cmap="YlGnBu")
```

Out[23]: <Axes: >



In [24]:
```
sns.boxplot(dataset)
```

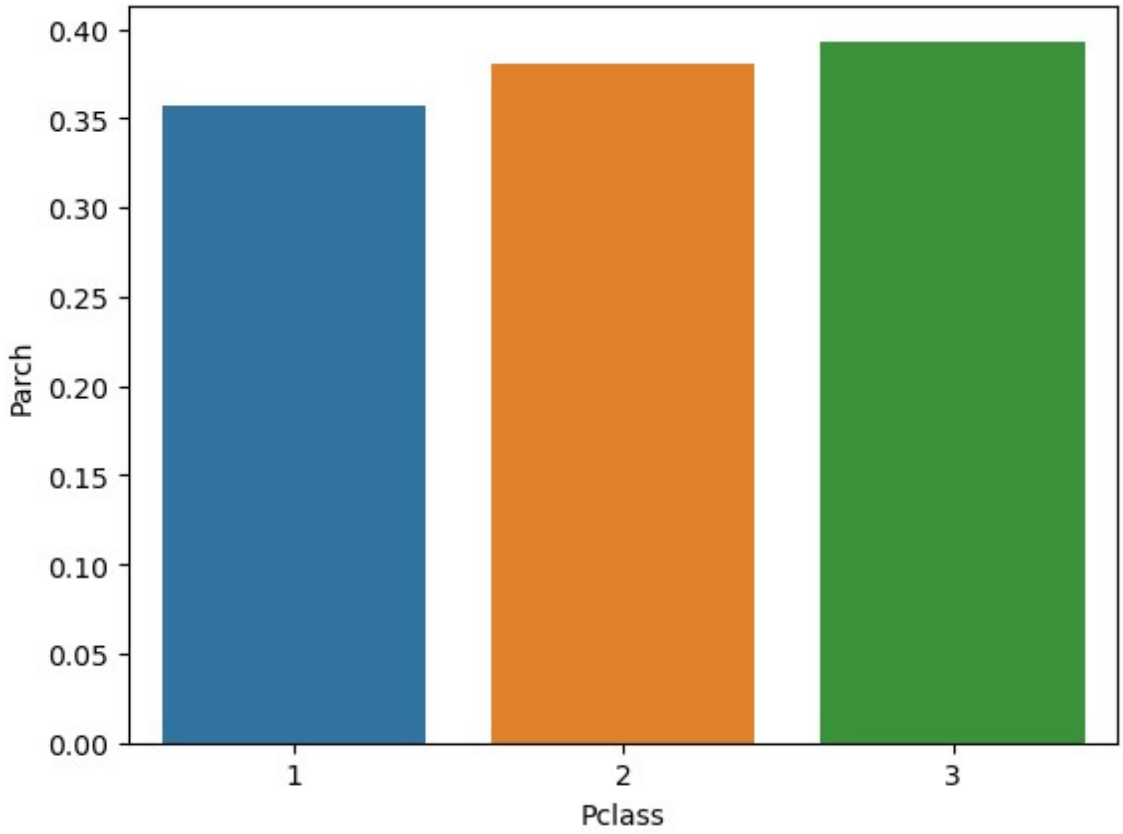
Out[24]: <Axes: >



In [25]:
```
sns.barplot(data=dataset,x="Pclass",y="Parch",ci=None)
```

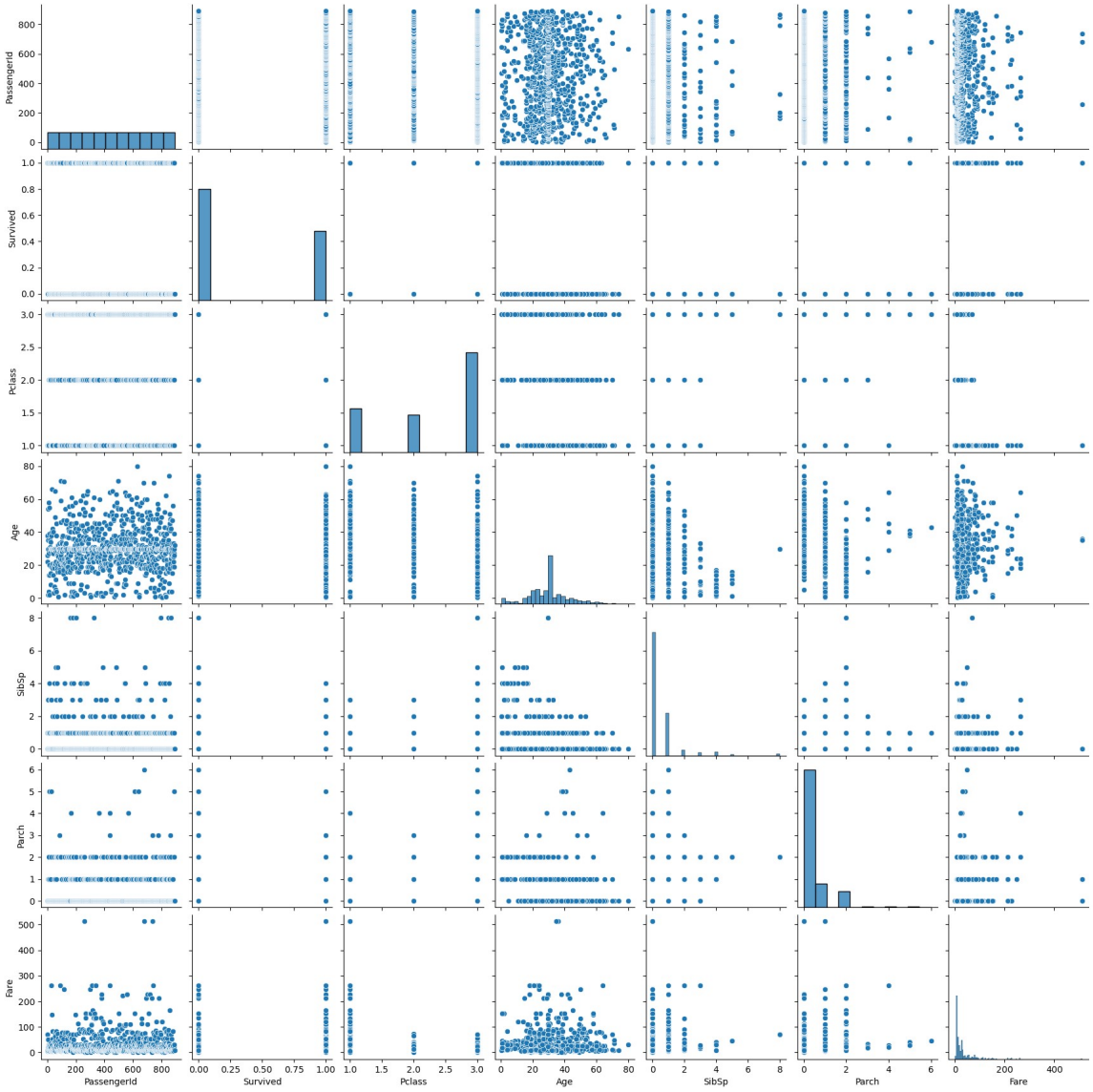
```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\2682752037.py:1: FutureWarni
ng:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.barplot(data=dataset,x="Pclass",y="Parch",ci=None)
```

Out[25]:
```
<Axes: xlabel='Pclass', ylabel='Parch'>
```



In [26]:
```
sns.pairplot(dataset)
```

Out[26]:
```
<seaborn.axisgrid.PairGrid at 0x2b3a6695150>
```



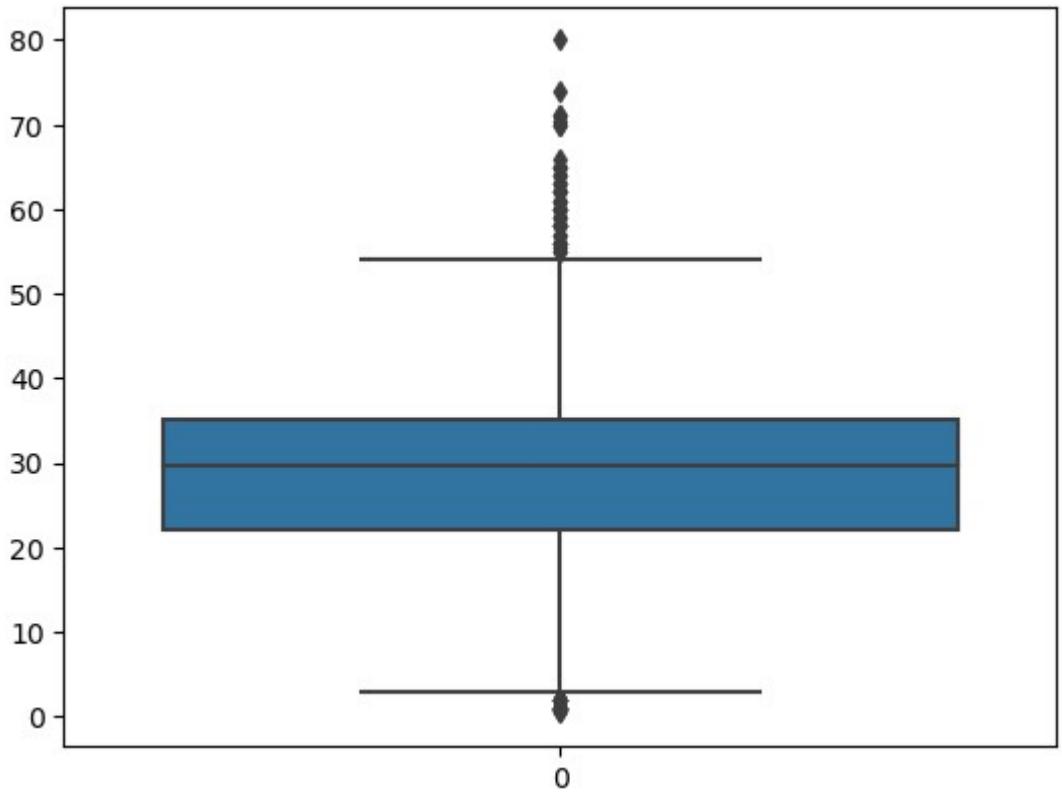
5.outlier detection

In [ ]:

In [ ]:

In [27]:
```
sns.boxplot(dataset.Age)
```

Out[27]: <Axes: >



In [28]:
```
q1=dataset.Age.quantile(0.25)
q3=dataset.Age.quantile(0.75)
q2=dataset.Age.quantile(0.50)
```

In [29]:
```
q1
```

Out[29]: 22.0

In [30]:
```
q3
```

Out[30]: 35.0

In [31]:
```
q2
```

Out[31]: 29.69911764705882

In [32]:
```
IQR=q3-q1
IQR
```

Out[32]: 13.0

In [33]:
```
upper_limit=q3+1.5*IQR
upper_limit
```

Out[33]: 54.5

In [34]:
```
lower_limit=q1-1.5*IQR
lower_limit
```

Out[34]:
```
2.5
```

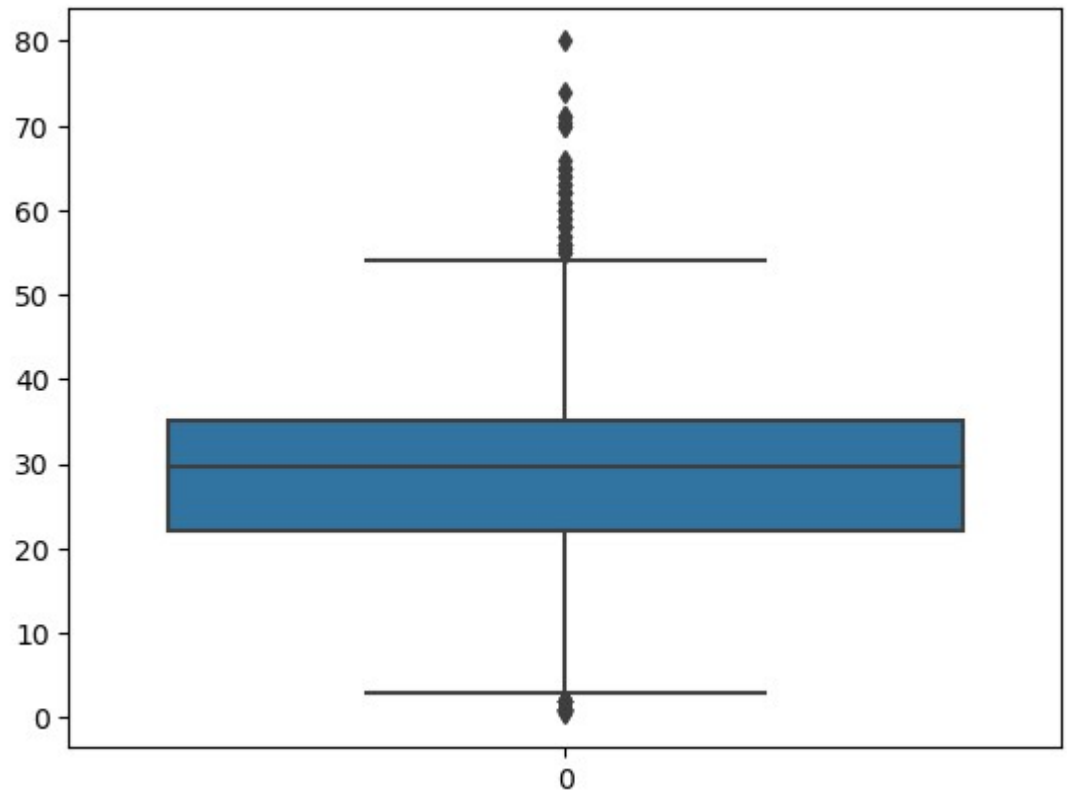
In [35]:
```
dataset.median()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\4167803218.py:1: FutureWarni
ng: The default value of numeric_only in DataFrame.median is deprecated. I
n a future version, it will default to False. In addition, specifying 'num
eric_only=None' is deprecated. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  dataset.median()
```

Out[35]:
```
PassengerId    446.000000
Survived         0.000000
Pclass           3.000000
Age             29.699118
SibSp            0.000000
Parch            0.000000
Fare            14.454200
dtype: float64
```

In [36]:
```
sns.boxplot(dataset.Age)
```

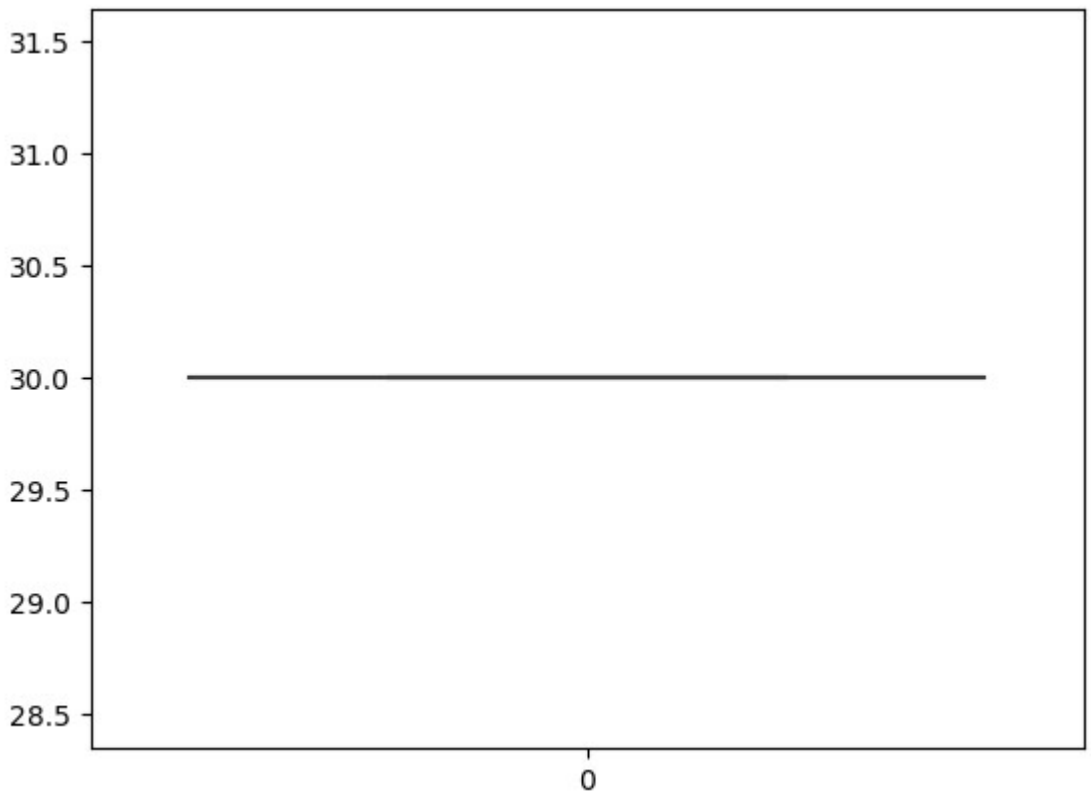
Out[36]:
```
<Axes: >
```



In [42]:
```
dataset['Age']=np.where(dataset['Age']>upper_limit,30,dataset['Age'])
```

In [43]:
```
sns.boxplot(dataset.Age)
```

Out[43]: <Axes: >



In [ ]:
```

```

In [46]:
```
X.shape
```

Out[46]: (891, 10)

In [47]:
```
q1=dataset.Fare.quantile(0.25)
q3=dataset.Fare.quantile(0.75)
q2=dataset.Fare.quantile(0.50)
```

In [48]:
```
q1
```

Out[48]: 7.9104

In [49]:
```
q2
```

Out[49]: 14.4542

In [50]:
```
q3
```

Out[50]: 31.0

In [51]:
```
IQR=q3-q1
IQR
```

Out[51]: 23.0896

In [52]:
```
upper_limit=q3+1.5*IQR
upper_limit
```

Out[52]:
```
65.6344
```

In [53]:
```
lower_limit=q1-1.5*IQR
lower_limit
```

Out[53]:
```
-26.724
```

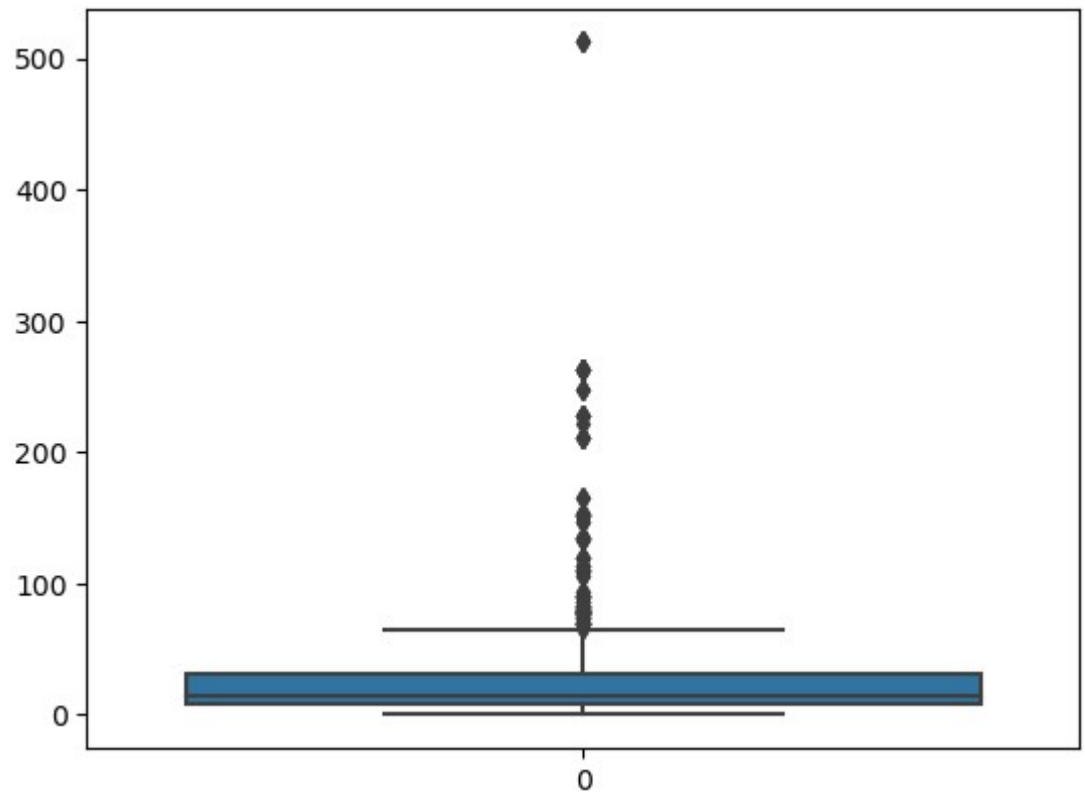
In [54]:
```
dataset.median()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\4167803218.py:1: FutureWarni
ng: The default value of numeric_only in DataFrame.median is deprecated. I
n a future version, it will default to False. In addition, specifying 'num
eric_only=None' is deprecated. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  dataset.median()
```

Out[54]:
```
PassengerId    446.0000
Survived         0.0000
Pclass           3.0000
Age             30.0000
SibSp            0.0000
Parch            0.0000
Fare            14.4542
dtype: float64
```

In [55]:
```
sns.boxplot(dataset.Fare)
```

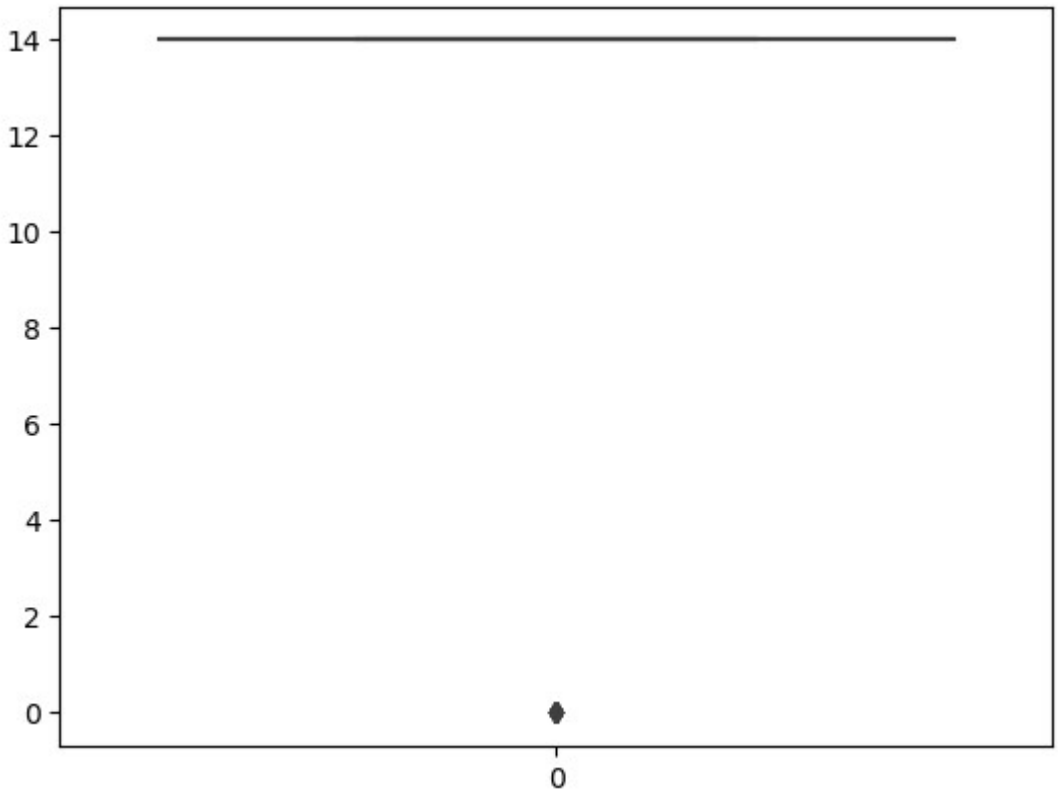
Out[55]:
```
<Axes: >
```



In [76]:
```
dataset['Fare']=np.where(dataset['Fare']>upper_limit,14,dataset['Fare'])
```

In [77]:
```
sns.boxplot(dataset.Fare)
```

Out[77]: <Axes: >



In [59]:
```
q1=dataset.Parch.quantile(0.25)
q3=dataset.Parch.quantile(0.75)
q2=dataset.Parch.quantile(0.50)
```

In [60]:
```
q1
```

Out[60]: 0.0

In [61]:
```
q2
```

Out[61]: 0.0

In [62]:
```
q3
```

Out[62]: 0.0

In [70]:
```
IQR=q3-q1
IQR
```

Out[70]: 0.0

In [71]:
```
upper_limit=q3+1.5*IQR
upper_limit
```

Out[71]: 0.0

In [72]:
```
lower_limit=q1-1.5*IQR
lower_limit
```

Out[72]:
```
0.0
```

In [73]:
```
dataset.median()
```

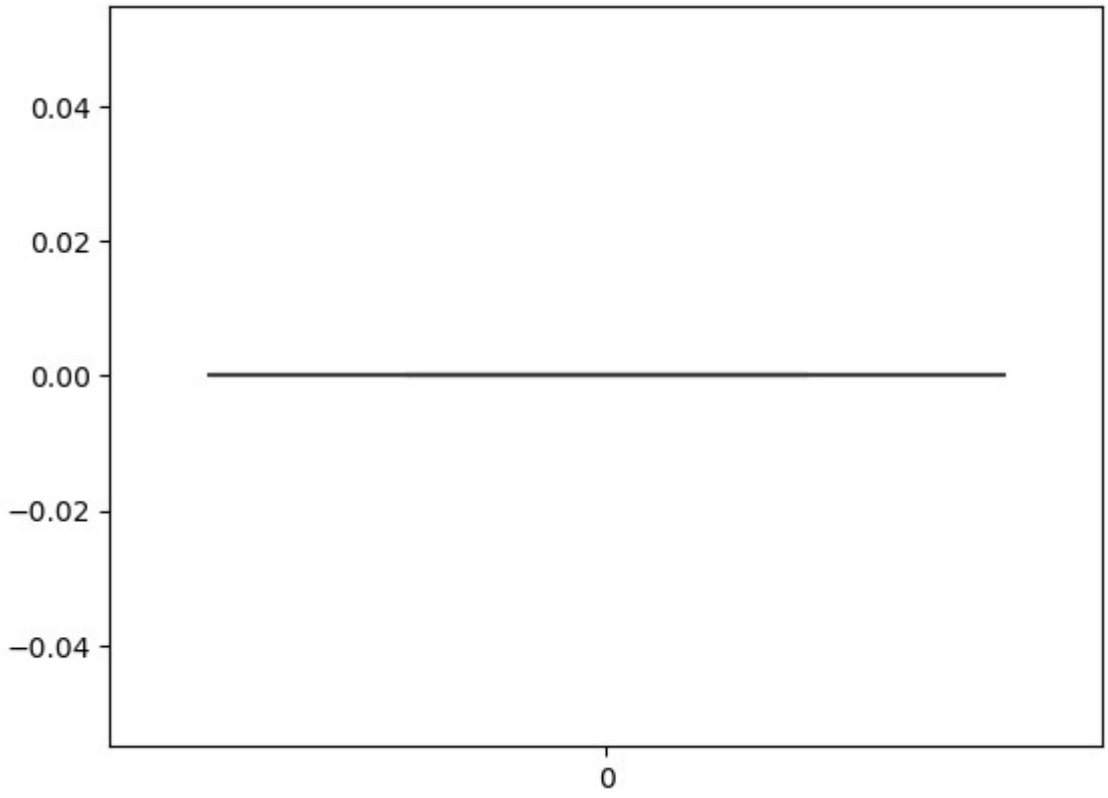
```
C:\Users\HP\AppData\Local\Temp\ipykernel_3208\4167803218.py:1: FutureWarni
ng: The default value of numeric_only in DataFrame.median is deprecated. I
n a future version, it will default to False. In addition, specifying 'num
eric_only=None' is deprecated. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  dataset.median()
```

Out[73]:
```
PassengerId    446.0000
Survived         0.0000
Pclass           3.0000
Age             30.0000
SibSp            0.0000
Parch            0.0000
Fare            14.4542
dtype: float64
```

In [74]:
```
dataset['Parch']=np.where(dataset['Parch']>upper_limit,0,dataset['Parch'])
```

In [75]:
```
sns.boxplot(dataset.Parch)
```

Out[75]:
```
<Axes: >
```



In [78]:
```
q1=dataset.SibSp.quantile(0.25)
q3=dataset.SibSp.quantile(0.75)
q2=dataset.SibSp.quantile(0.50)
```

In [79]:
```
q1
```

Out[79]: 0.0

In [80]:
```
q2
```

Out[80]: 0.0

In [81]:
```
q3
```

Out[81]: 1.0

In [82]:
```
IQR=q3-q1
IQR
```

Out[82]: 1.0

In [83]:
```
upper_limit=q3+1.5*IQR
upper_limit
```

Out[83]: 2.5

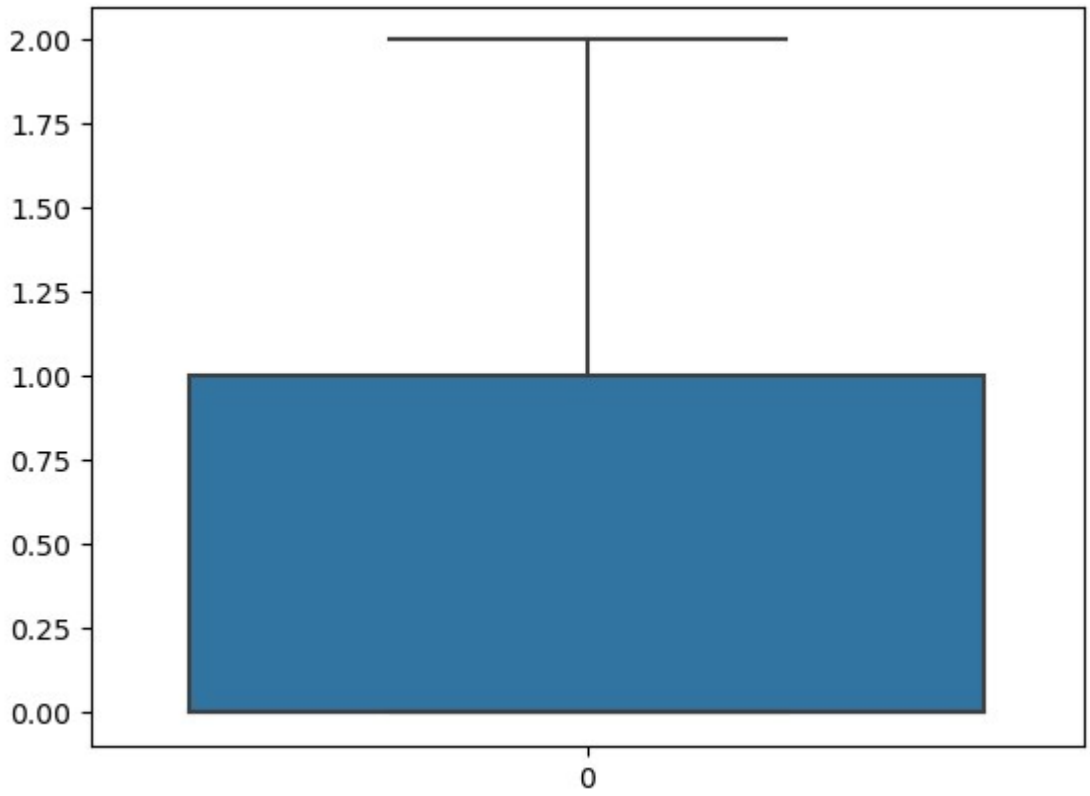
In [84]:
```
lower_limit=q1-1.5*IQR
lower_limit
```

Out[84]: -1.5

In [85]:
```
dataset['SibSp']=np.where(dataset['SibSp']>upper_limit,0,dataset['SibSp'])
```

In [86]:
```
sns.boxplot(dataset.SibSp)
```

Out[86]:
```
<Axes: >
```



6 ENCODER

In [87]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

In [100]:
```
le
```

Out[100]:
```
▾ LabelEncoder
LabelEncoder()
```

In [89]:
```
dataset.head()
```

Out[89]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Er
0	1	0	3	Braund, Mr. Owen Harris	male	30.0	1	0	A/5 21171	14.0	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	30.0	1	0	PC 17599	14.0	
2	3	1	3	Heikkinen, Miss. Laina	female	30.0	0	0	STON/O2. 3101282	14.0	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	30.0	1	0	113803	14.0	
4	5	0	3	Allen, Mr. William Henry	male	30.0	0	0	373450	14.0	

In [92]:
```
dataset["Name"]=le.fit_transform(dataset["Name"])
```

In [93]:
```
dataset.head()
```

Out[93]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embar
0	1	0	3	108	male	30.0	1	0	A/5 21171	14.0	
1	2	1	1	190	female	30.0	1	0	PC 17599	14.0	
2	3	1	3	353	female	30.0	0	0	STON/O2. 3101282	14.0	
3	4	1	1	272	female	30.0	1	0	113803	14.0	
4	5	0	3	15	male	30.0	0	0	373450	14.0	

In [94]:
```
dataset["Sex"]=le.fit_transform(dataset["Sex"])
```

In [95]:
```
dataset.head()
```

Out[95]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarke
0	1	0	3	108	1	30.0	1	0	A/5 21171	14.0	
1	2	1	1	190	0	30.0	1	0	PC 17599	14.0	
2	3	1	3	353	0	30.0	0	0	STON/O2. 3101282	14.0	
3	4	1	1	272	0	30.0	1	0	113803	14.0	
4	5	0	3	15	1	30.0	0	0	373450	14.0	

In [96]:
```
dataset["Ticket"]=le.fit_transform(dataset["Ticket"])
```

In [97]:
```
dataset.head()
```

Out[97]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	108	1	30.0	1	0	523	14.0	S
1	2	1	1	190	0	30.0	1	0	596	14.0	C
2	3	1	3	353	0	30.0	0	0	669	14.0	S
3	4	1	1	272	0	30.0	1	0	49	14.0	S
4	5	0	3	15	1	30.0	0	0	472	14.0	S

In [98]:
```
dataset["Embarked"]=le.fit_transform(dataset["Embarked"])
```

In [99]:
```
dataset.head()
```

Out[99]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	108	1	30.0	1	0	523	14.0	2
1	2	1	1	190	0	30.0	1	0	596	14.0	0
2	3	1	3	353	0	30.0	0	0	669	14.0	2
3	4	1	1	272	0	30.0	1	0	49	14.0	2
4	5	0	3	15	1	30.0	0	0	472	14.0	2

7splitting data

In [116]:
```
x=dataset.iloc[0:5,2:12]
x.head()
```

Out[116]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	3	108	1	30.0	1	0	523	14.0	2
1	1	190	0	30.0	1	0	596	14.0	0
2	3	353	0	30.0	0	0	669	14.0	2
3	1	272	0	30.0	1	0	49	14.0	2
4	3	15	1	30.0	0	0	472	14.0	2

In [120]:
```
y=dataset.iloc[0:5,0:2]
```

In [121]:
```
y
```

Out[121]:

	PassengerId	Survived
0	1	0
1	2	1
2	3	1
3	4	1
4	5	0

In [122]:
```
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_sta
```

In [132]:
```
x_train
```

Out[132]:
```
array([[-0.70710678,  0.28921904, -0.70710678,  0.        ,  0.70710678,
         0.        ,  0.95515264,  0.        , -1.41421356],
       [-0.70710678,  1.05425005, -0.70710678,  0.        ,  0.70710678,
         0.        , -1.38077208,  0.        ,  0.70710678],
       [ 1.41421356, -1.34346909,  1.41421356,  0.        , -1.41421356,
         0.        ,  0.42561943,  0.        ,  0.70710678]])
```

In [ ]:
```

```

In [123]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[123]:
```
((3, 9), (2, 9), (3, 2), (2, 2))
```

8.feature scaling

In [124]:
```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
```

In [125]:
```
x_train=sc.fit_transform(x_train)
x_test=sc.fit_transform(x_test)
```

In [126]:
```
x_train
```

Out[126]:
```
array([[-0.70710678,  0.28921904, -0.70710678,  0.        ,  0.70710678,
         0.        ,  0.95515264,  0.        , -1.41421356],
       [-0.70710678,  1.05425005, -0.70710678,  0.        ,  0.70710678,
         0.        , -1.38077208,  0.        ,  0.70710678],
       [ 1.41421356, -1.34346909,  1.41421356,  0.        , -1.41421356,
         0.        ,  0.42561943,  0.        ,  0.70710678]])
```

In [127]:
```
x_test
```

Out[127]:
```
array([[ 0.,  1., -1.,  0., -1.,  0.,  1.,  0.,  0.],
       [ 0., -1.,  1.,  0.,  1.,  0., -1.,  0.,  0.]])
```

In [ ]:
```

```