Vishal Kumar Singh 21BCE8647 VIT-AP

Importing necessary Libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Importing the dataset

```
df=pd.read_csv("Titanic-Dataset.csv")
```

```
df.head()
```

	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	female	38.0	1	0	PC 17599	71.2833

```
df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
df.corr()
```

```
<ipython-input-5-2f6f6606aa2c>:1: FutureWarning: The default value of numeric_only in Da
  df.corr()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1.000000	-0.005007	-0.035144	0.036847	-0.057527	-0.001652	0.012658
Survived	-0.005007	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307
Pclass	-0.035144	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500
Age	0.036847	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067
SibSp	-0.057527	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651
Parch	-0.001652	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225
Fare	0.012658	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000

```
df.corr().Fare.sort_values(ascending=False)
```

```
<ipython-input-7-f51f352aac84>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version,
  df.corr().Fare.sort_values(ascending=False)
Fare           1.000000
Survived       0.257307
Parch          0.216225
```

```
SibSp          0.159651
Age            0.096067
PassengerId    0.012658
Pclass        -0.549500
Name: Fare, dtype: float64
```

Checking for null values

```
df.isnull().any()
```

```
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
```

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

```
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
```

```
df.Age.nunique()
```

```
88
```

```
df.Age.unique()
```

```
array([22.  , 38.  , 26.  , 35.  ,   nan, 54.  ,  2.  , 27.  , 14.  ,
        4.  , 58.  , 20.  , 39.  , 55.  , 31.  , 34.  , 15.  , 28.  ,
        8.  , 19.  , 40.  , 66.  , 42.  , 21.  , 18.  ,  3.  ,  7.  ,
       49.  , 29.  , 65.  , 28.5 ,  5.  , 11.  , 45.  , 17.  , 32.  ,
       16.  , 25.  ,  0.83, 30.  , 33.  , 23.  , 24.  , 46.  , 59.  ,
       71.  , 37.  , 47.  , 14.5 , 70.5 , 32.5 , 12.  ,  9.  , 36.5 ,
       51.  , 55.5 , 40.5 , 44.  ,  1.  , 61.  , 56.  , 50.  , 36.  ,
       45.5 , 20.5 , 62.  , 41.  , 52.  , 63.  , 23.5 ,  0.92, 43.  ,
       60.  , 10.  , 64.  , 13.  , 48.  ,  0.75, 53.  , 57.  , 80.  ,
       70.  , 24.5 ,  6.  ,  0.67, 30.5 ,  0.42, 34.5 , 74.  ])
```

```
df.Cabin.nunique()
```

```
147
```

```
df.Cabin.unique()
```

```
array([nan, 'C85', 'C123', 'E46', 'G6', 'C103', 'D56', 'A6',
       'C23 C25 C27', 'B78', 'D33', 'B30', 'C52', 'B28', 'C83', 'F33',
       'F G73', 'E31', 'A5', 'D10 D12', 'D26', 'C110', 'B58 B60', 'E101',
       'F E69', 'D47', 'B86', 'F2', 'C2', 'E33', 'B19', 'A7', 'C49', 'F4',
       'A32', 'B4', 'B80', 'A31', 'D36', 'D15', 'C93', 'C78', 'D35',
       'C87', 'B77', 'E67', 'B94', 'C125', 'C99', 'C118', 'D7', 'A19',
       'B49', 'D', 'C22 C26', 'C106', 'C65', 'E36', 'C54',
       'B57 B59 B63 B66', 'C7', 'E34', 'C32', 'B18', 'C124', 'C91', 'E40',
       'T', 'C128', 'D37', 'B35', 'E50', 'C82', 'B96 B98', 'E10', 'E44',
       'A34', 'C104', 'C111', 'C92', 'E38', 'D21', 'E12', 'E63', 'A14',
       'B37', 'C30', 'D20', 'B79', 'E25', 'D46', 'B73', 'C95', 'B38',
       'B39', 'B22', 'C86', 'C70', 'A16', 'C101', 'C68', 'A10', 'E68',
       'B41', 'A20', 'D19', 'D50', 'D9', 'A23', 'B50', 'A26', 'D48',
       'E58', 'C126', 'B71', 'B51 B53 B55', 'D49', 'B5', 'B20', 'F G63',
       'C62 C64', 'E24', 'C90', 'C45', 'E8', 'B101', 'D45', 'C46', 'D30',
       'E121', 'D11', 'E77', 'F38', 'B3', 'D6', 'B82 B84', 'D17', 'A36',
       'B102', 'B69', 'E49', 'C47', 'D28', 'E17', 'A24', 'C50', 'B42',
       'C148'], dtype=object)
```

```
df.Embarked.nunique()
```

```
3
```

```
df.Embarked.unique()
```

```
array(['S', 'C', 'Q', nan], dtype=object)
```

Handling the null values

```
df["Age"].fillna(df["Age"].mean(),inplace=True)
```

```
df["Cabin"].fillna(df["Cabin"].mode(),inplace=True)
```

```
df["Embarked"].fillna(df["Embarked"].mode(),inplace=True)
```

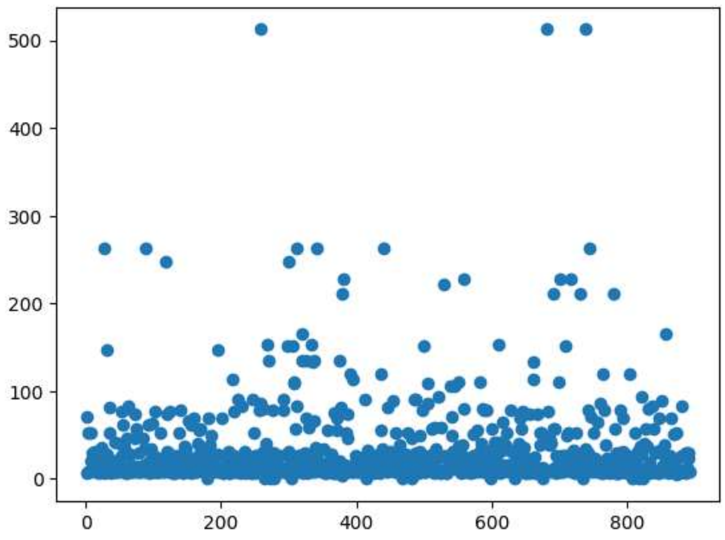
```
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embark
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	B96 B98	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th	female	38.0	1	0	PC 17599	71.2833	C85	

Data Visualization

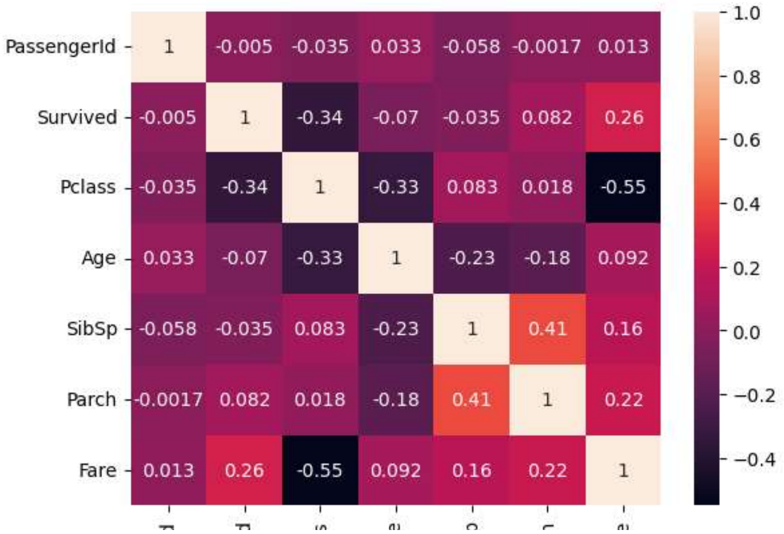
```
plt.scatter(df["PassengerId"],df["Fare"])
```

```
<matplotlib.collections.PathCollection at 0x7f44570cf1f0>
```



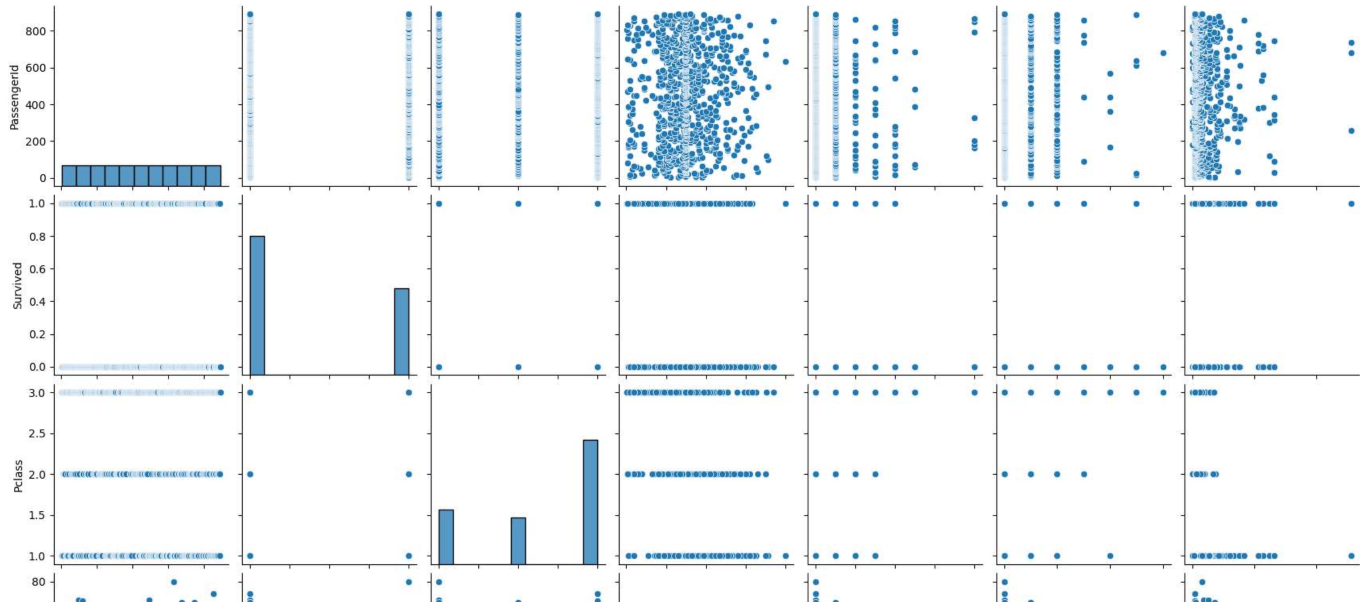
```
sns.heatmap(df.corr(),annot=True)
```

```
<ipython-input-29-8df7bcac526d>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future versior
  sns.heatmap(df.corr(),annot=True)
<Axes: >
```



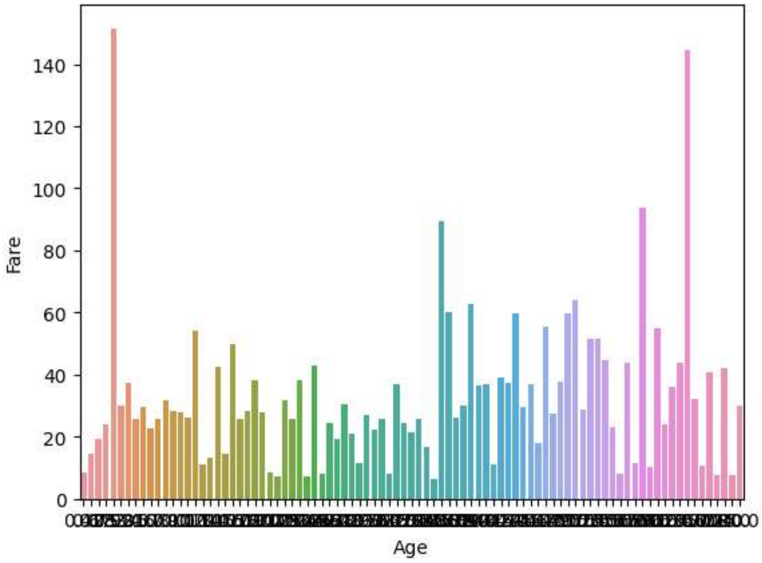
```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x7f4456bf34f0>
```



```
sns.barplot(x=df['Age'],y=df['Fare'],ci=0)
```

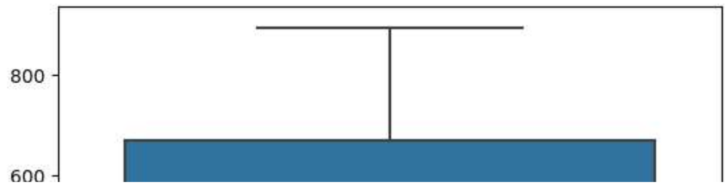
```
<ipython-input-31-8e72dcd4708e>:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

  sns.barplot(x=df['Age'],y=df['Fare'],ci=0)
<Axes: xlabel='Age', ylabel='Fare'>
```



Outlier Detection

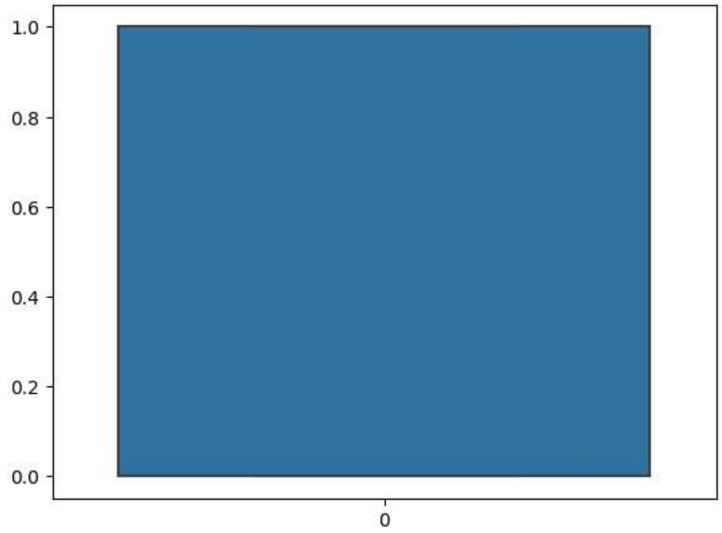
```
sns.boxplot(df["PassengerId"])
```

```
<Axes: >
```



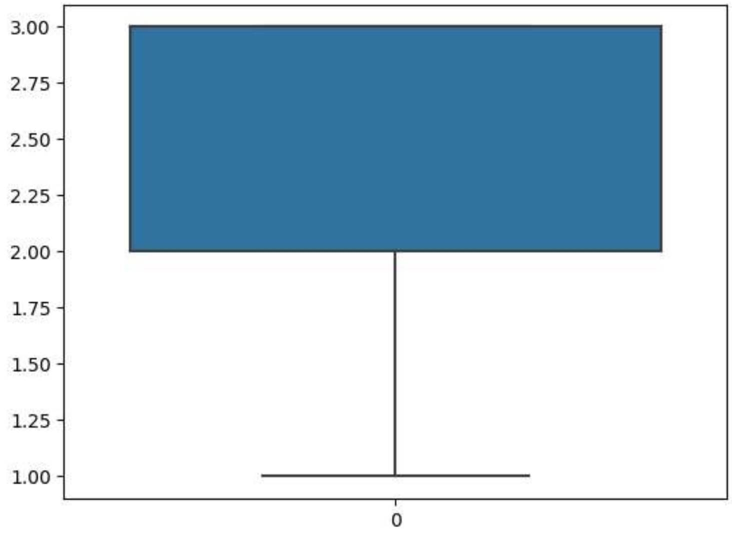
```
sns.boxplot(df["Survived"])
```

```
<Axes: >
```



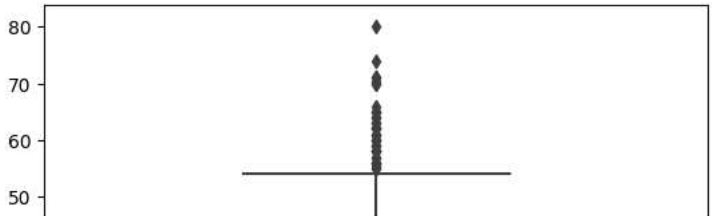
```
sns.boxplot(df["Pclass"])
```

```
<Axes: >
```



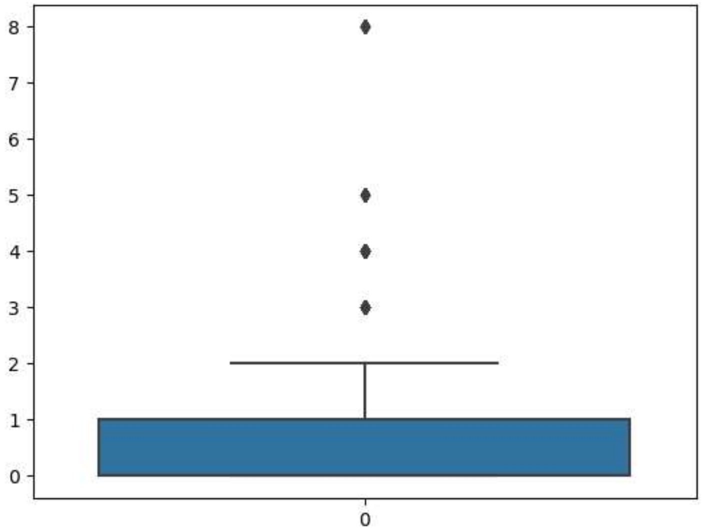
```
sns.boxplot(df["Age"])
```

```
<Axes: >
```



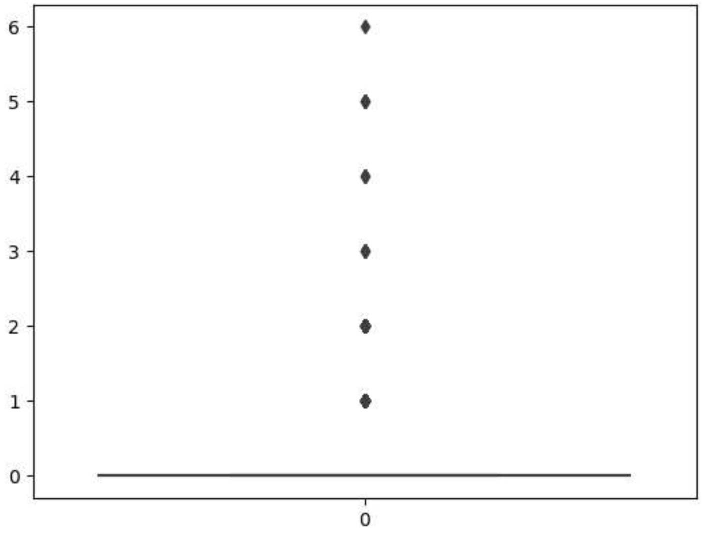
```
sns.boxplot(df["SibSp"])
```

```
<Axes: >
```



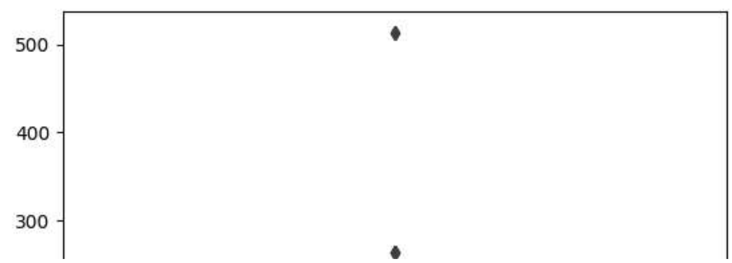
```
sns.boxplot(df["Parch"])
```

```
<Axes: >
```



```
sns.boxplot(df["Fare"])
```

```
<Axes: >
```



Spliting dependent and independent variables

```
x=df.drop(columns=["Fare"],axis=1)
```

```
x.shape
```

```
(891, 11)
```

```
type(x)
```

```
pandas.core.frame.DataFrame
```

```
y=df["Fare"]
```

```
y.head()
```

```
0     7.2500
1    71.2833
2     7.9250
3    53.1000
4     8.0500
Name: Fare, dtype: float64
```

Encoding

```
from sklearn.preprocessing import LabelEncoder
```

```
le=LabelEncoder()
```

```
x["Embarked"]=le.fit_transform(x["Embarked"])
```

```
x["Cabin"]=le.fit_transform(x["Cabin"])
```

```
x.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	47	2
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	81	0
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	145	2
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	55	2

```
x["Ticket"]=le.fit_transform(x["Ticket"])
```

```
x["Sex"]=le.fit_transform(x["Sex"])
```

```
x.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	1	22.0	1	0	523	47	2
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	596	81	0
2	3	1	3	Heikkinen, Miss. Laina	0	26.0	0	0	669	145	2

```
print(le.classes_)
```

```
['female' 'male']
```

```
mapping=dict(zip(le.classes_,range(len(le.classes_))))
```

```
mapping
```

```
{'female': 0, 'male': 1}
```

0s completed at 9:11 PM