ASSIGNMENT 3

GUDLA NISHIKA

21BCE8562

VIT-AP University

Data Preprocessing.

1. Import the Libraries.
2. Importing the dataset.
3. Checking for Null Values.
4. Data Visualization.
5. Outlier Detection
6. Splitting Dependent and Independent variables
7. Encoding
8. Feature Scaling.
9. Splitting Data into Train and Test.

1. Import the Libraries.

In [ ]:

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

2. Importing the dataset.

In [ ]:

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

In [ ]:

```
df
```

Out[ ]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticke
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 2117
1	2	1	1	Cumings, Mrs. John Bradley (Florence	female	38.0	1	0	PC 1759

				Briggs Th...					
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2 310128
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	11380
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	37345
...	...	...	...	...	...	...	...	...	.
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	21153
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	11205
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C 660
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	11136
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	37037

891 rows × 12 columns

In [ ]:

```
df.head()
```

Out[ ]:

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

				(Lily May Peel)					
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450

```
In [ ]: df.shape
```

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

```
In [ ]: df.describe()
```

Out[ ]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 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  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

```
In [ ]: df.corr(numeric_only=True)
```

Out[ ]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	
PassengerId	1.000000	-0.005007	-0.035144	0.036847	-0.057527	-0.001652	0.01
Survived	-0.005007	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.25
Pclass	-0.035144	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.54
Age	0.036847	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.09

SibSp	-0.057527	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.15
Parch	-0.001652	0.081629	0.018443	-0.189119	0.414838	1.000000	0.21
Fare	0.012658	0.257307	-0.549500	0.096067	0.159651	0.216225	1.00

```
In [ ]: df.corr(numeric_only=True).Survived.sort_values(ascending=False)
```

```
Out[ ]: Survived       1.000000
        Fare           0.257307
        Parch          0.081629
        PassengerId   -0.005007
        SibSp         -0.035322
        Age           -0.077221
        Pclass        -0.338481
        Name: Survived, dtype: float64
```

```
In [ ]: df.Survived.value_counts()
```

```
Out[ ]: 0    549
        1    342
        Name: Survived, dtype: int64
```

3. Checking for Null Values.

```
In [ ]: df.isnull().any()
```

```
Out[ ]: 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 [ ]: df.isnull().sum()
```

```
Out[ ]: 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 [ ]: df.Embarked.nunique()
```

```
Out[ ]: 3
```

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

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

```
In [ ]: df.Embarked.value_counts()
```

```
Out[ ]: S    644
        C    168
        Q     77
        Name: Embarked, dtype: int64
```

Null Values are present in Age,Cabin and Embarked. We need to handle null values to proceed to next step.

```
In [ ]: #median method
        df['Age'].fillna(df['Age'].median(),inplace=True)
```

```
In [ ]: #imputing method
        df['Cabin'].fillna('Unknown',inplace=True)
```

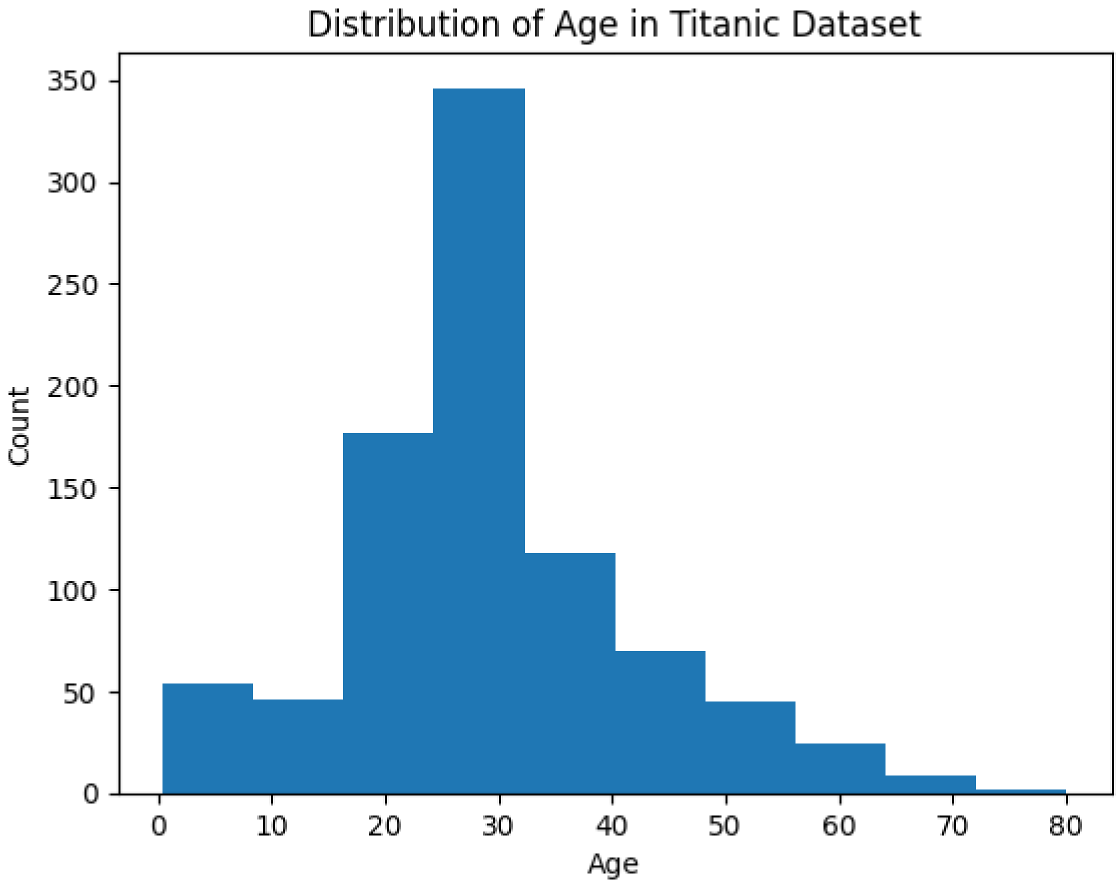
```
In [ ]: #mode method
        df['Embarked'].fillna(df['Embarked'].mode()[0],inplace=True)
```

```
In [ ]: df.isnull().sum() #All null values are sussessfully handled.
```

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

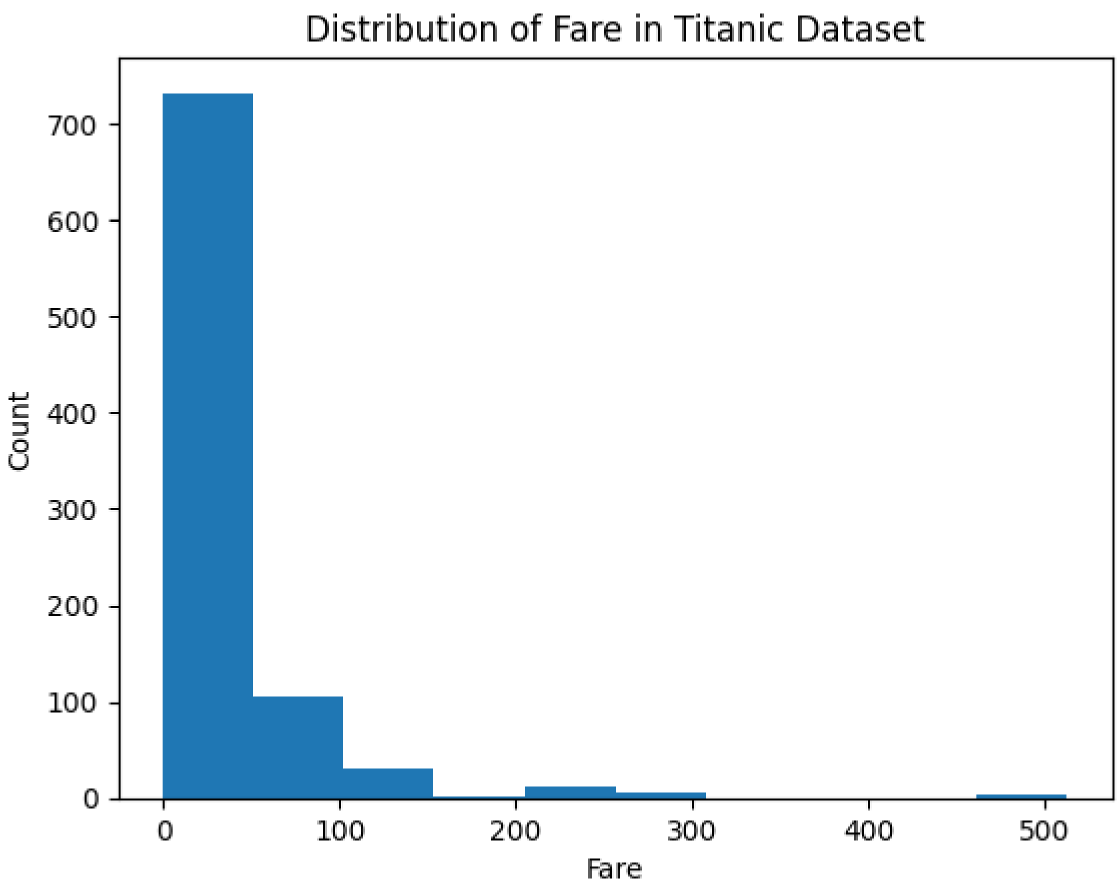
4. Data Visualization.

```
In [ ]: plt.hist(df['Age'])
        plt.xlabel('Age')
        plt.ylabel('Count')
        plt.title('Distribution of Age in Titanic Dataset')
        plt.show()
```



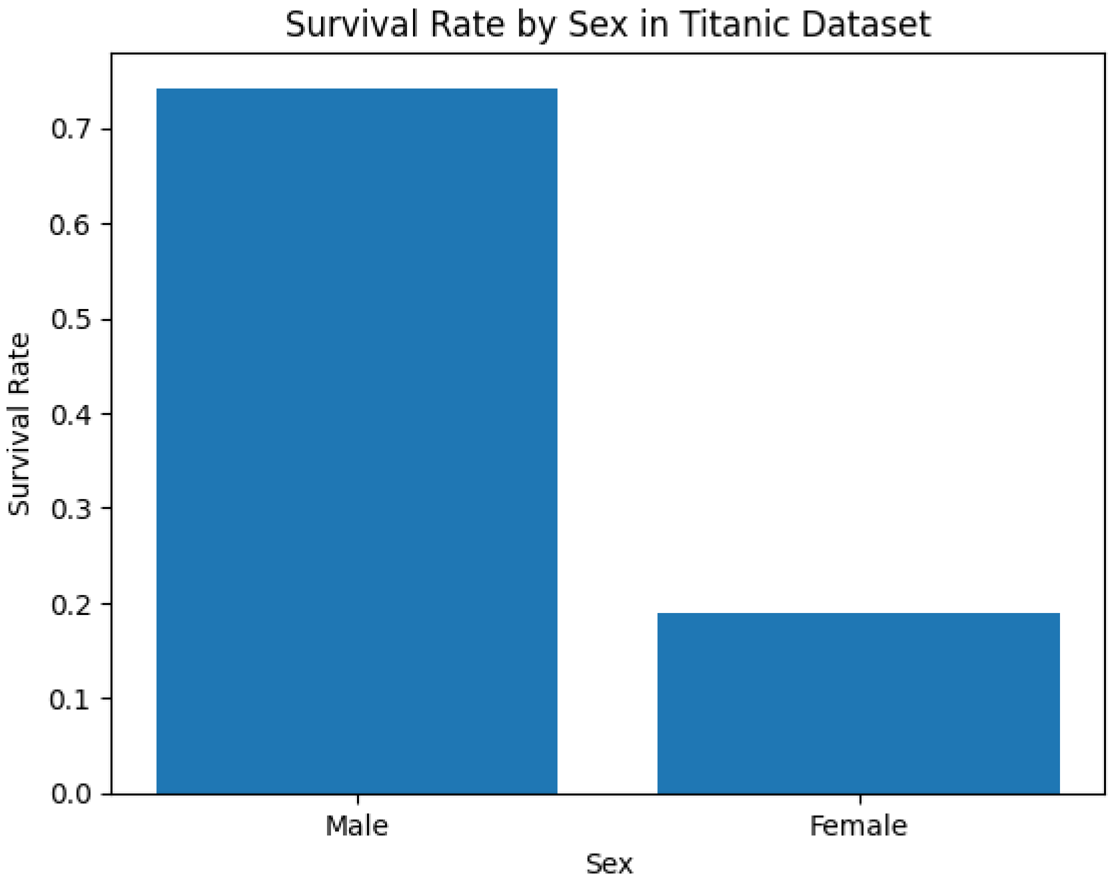
In [ ]:

```
plt.hist(df['Fare'])
plt.xlabel('Fare')
plt.ylabel('Count')
plt.title('Distribution of Fare in Titanic Dataset')
plt.show()
```



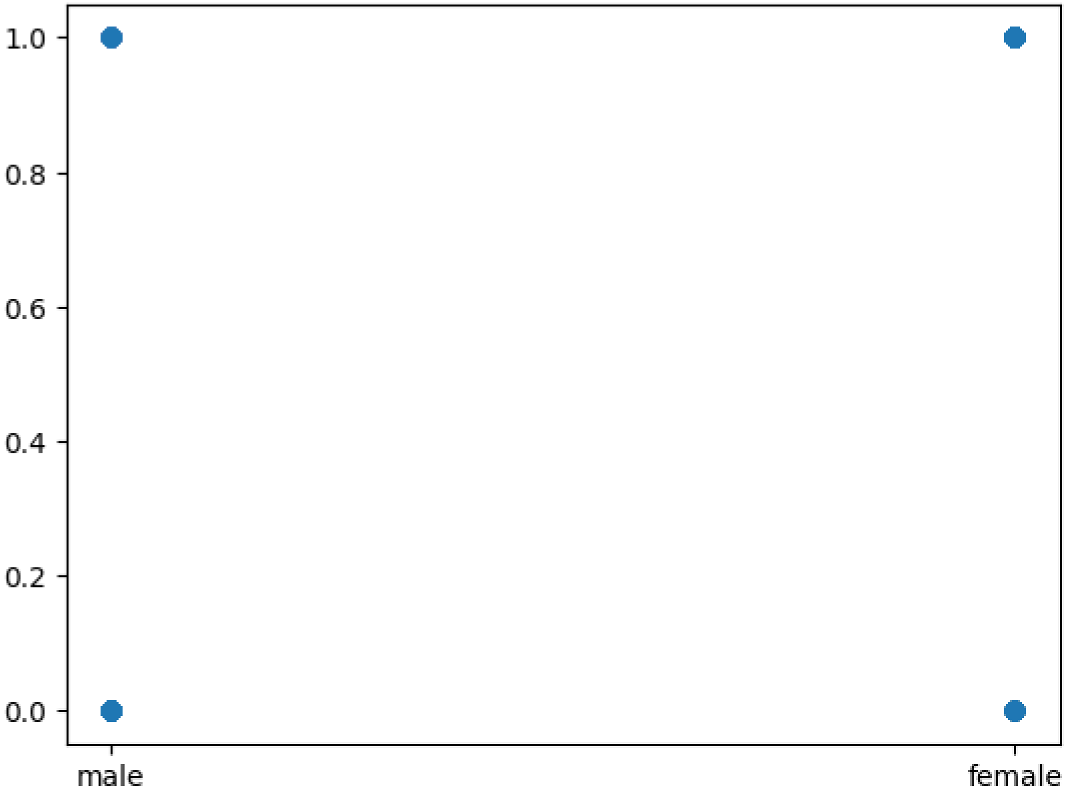
In [ ]:

```
plt.bar(['Male', 'Female'], df.groupby('Sex')['Survived'].mean())
plt.xlabel('Sex')
plt.ylabel('Survival Rate')
plt.title('Survival Rate by Sex in Titanic Dataset')
plt.show()
```



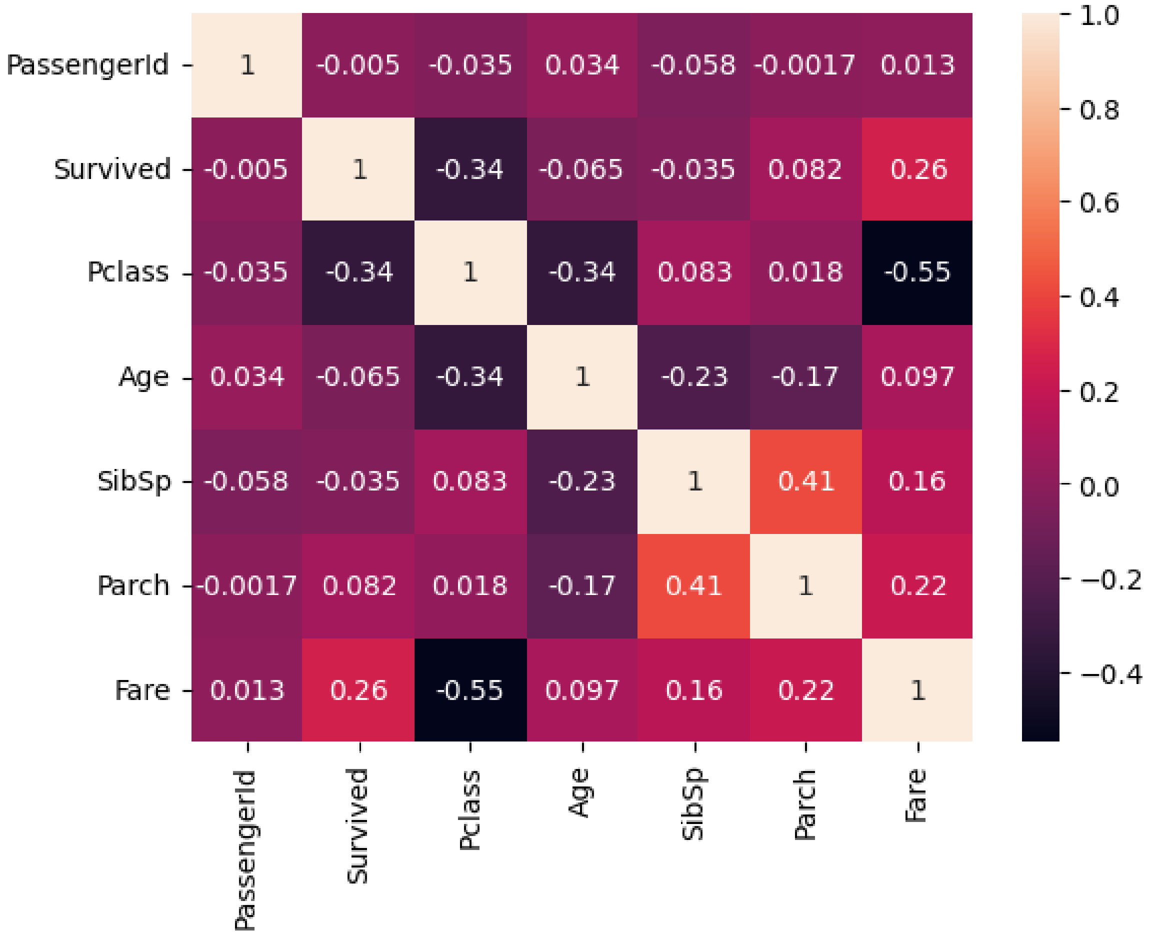
In [ ]:
```
plt.scatter(df["Sex"],df["Survived"])
```

Out[ ]:
```
<matplotlib.collections.PathCollection at 0x18cffc39d00>
```



In [ ]:
```
sns.heatmap(df.corr(numeric_only=True),annot=True)
```

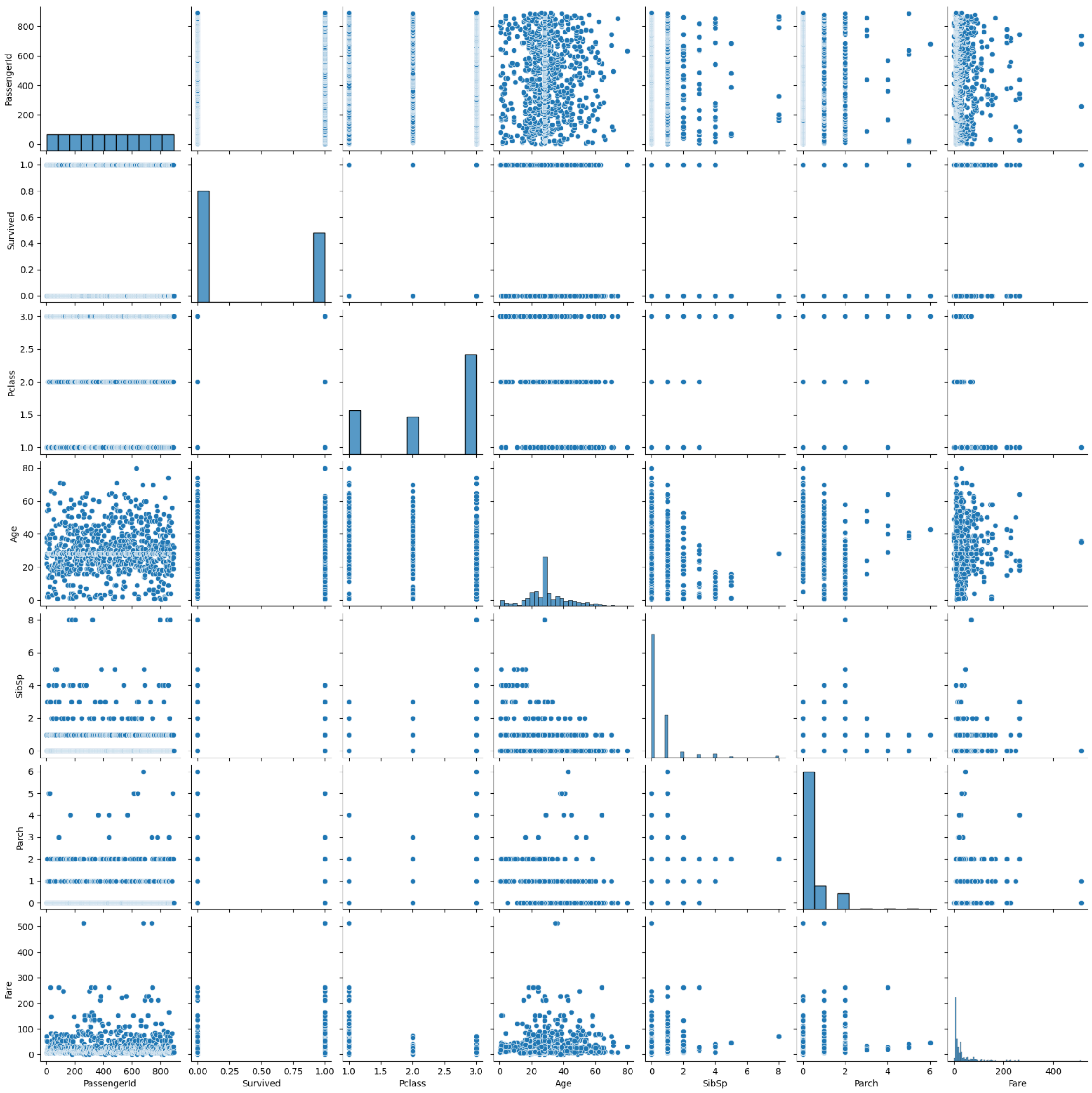
Out[ ]:
```
<Axes: >
```



	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1	-0.005	-0.035	0.034	-0.058	-0.0017	0.013
Survived	-0.005	1	-0.34	-0.065	-0.035	0.082	0.26
Pclass	-0.035	-0.34	1	-0.34	0.083	0.018	-0.55
Age	0.034	-0.065	-0.34	1	-0.23	-0.17	0.097
SibSp	-0.058	-0.035	0.083	-0.23	1	0.41	0.16
Parch	-0.0017	0.082	0.018	-0.17	0.41	1	0.22
Fare	0.013	0.26	-0.55	0.097	0.16	0.22	1

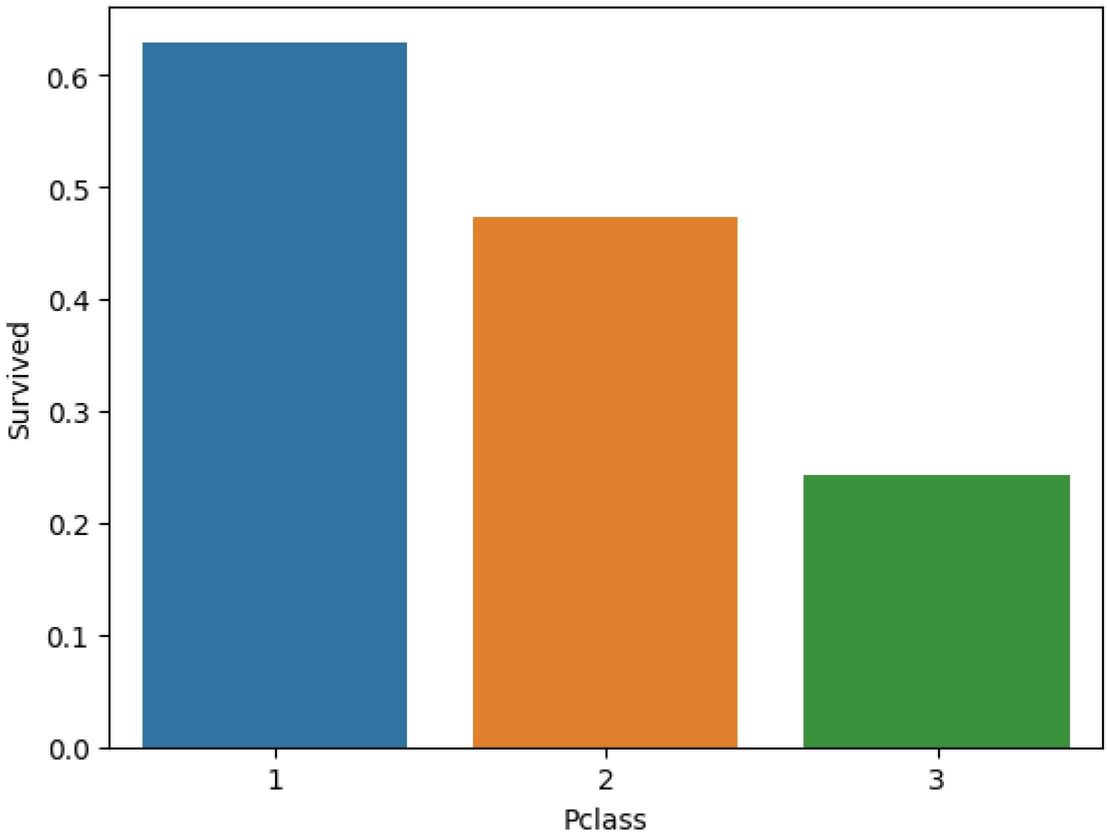
```
In [ ]: sns.pairplot(df)
```

```
Out[ ]: <seaborn.axisgrid.PairGrid at 0x18cff9f5f40>
```



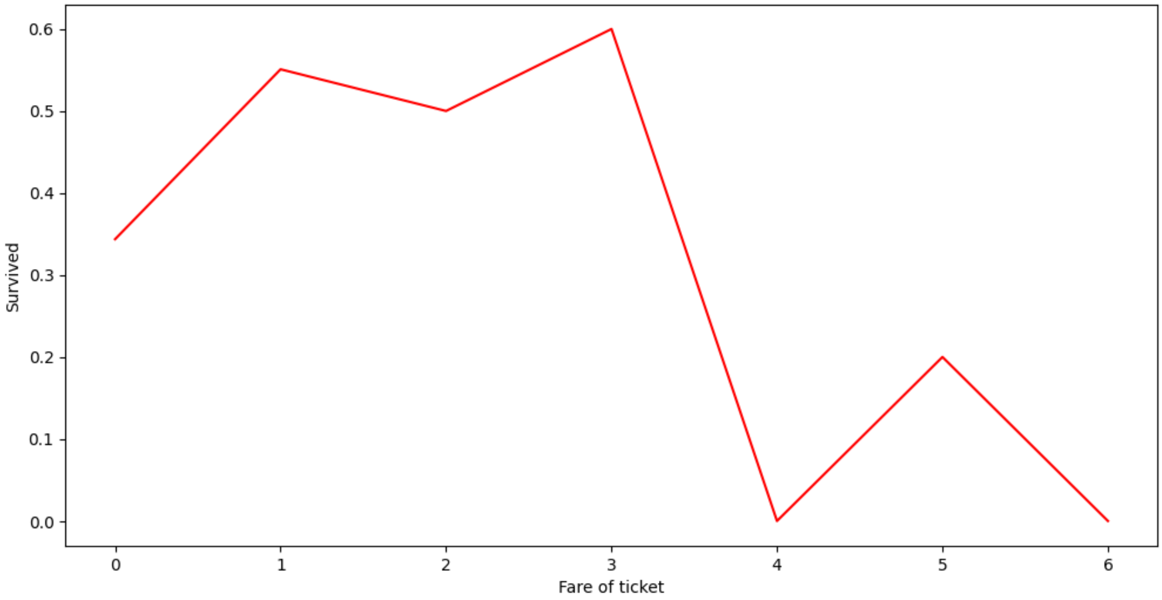
```
In [ ]: sns.barplot(x=df["Pclass"], y=df["Survived"], errorbar=('ci', 0))
```

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



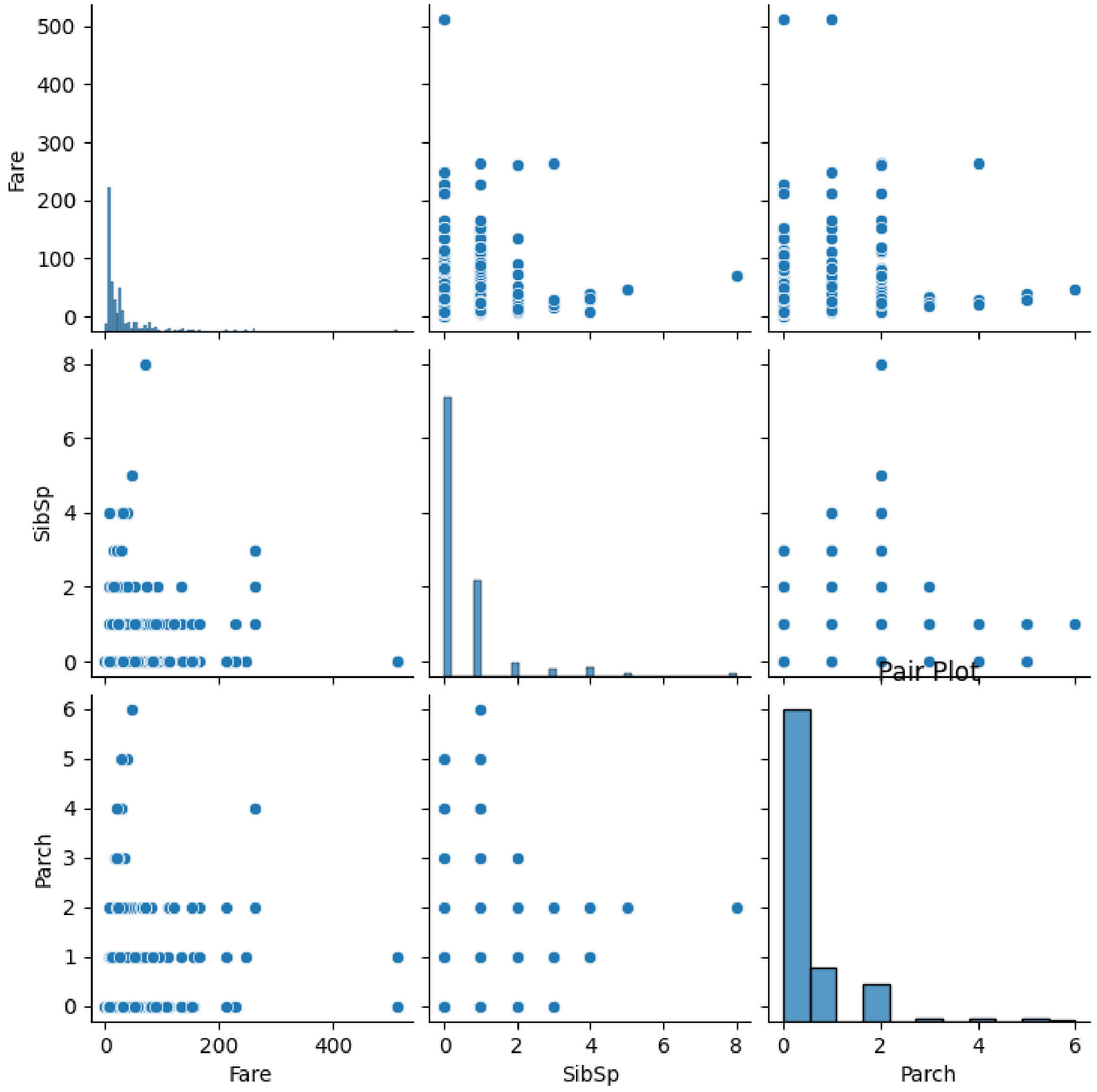
In [ ]:

```
plt.figure(figsize=(12,6))
sns.lineplot(x='Parch', y='Survived', data=df, errorbar=None, color = "re
plt.xlabel('Fare of ticket')
plt.ylabel('Survived')
plt.show()
```



In [ ]:

```
sns.pairplot(data=df[['Fare','SibSp','Parch']])
plt.title('Pair Plot')
plt.show()
```



5. Outlier Detection

In [ ]:

```
df.head()
```

Out[ ]:

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

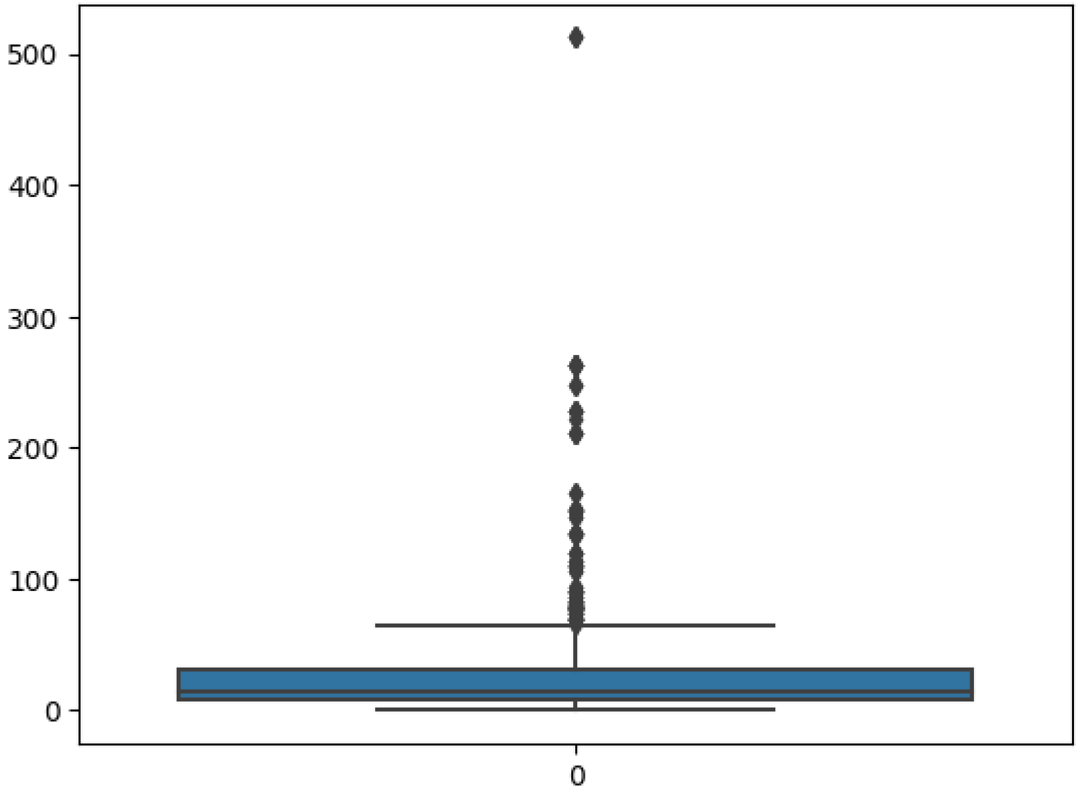
				(Lily May Peel)					
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450

In [ ]:

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

Out[ ]:

```
<Axes: >
```

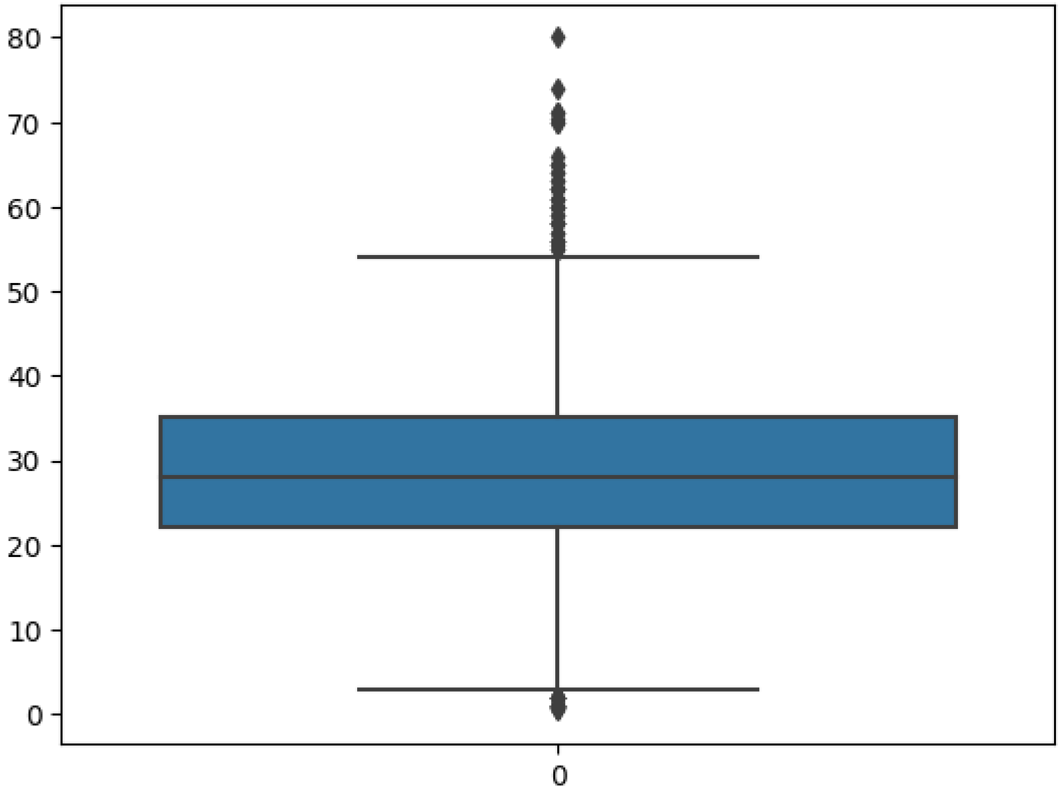


In [ ]:

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

Out[ ]:

```
<Axes: >
```



```
In [ ]: from scipy import stats
        from scipy.stats import zscore
```

Using Z-Score for Age

```
In [ ]: fare_zscore = stats.zscore(df.Age)
        fare_zscore
```

```
Out[ ]: 0     -0.565736
        1      0.663861
        2     -0.258337
        3      0.433312
        4      0.433312
                 ...
        886   -0.181487
        887   -0.796286
        888   -0.104637
        889   -0.258337
        890    0.202762
        Name: Age, Length: 891, dtype: float64
```

```
In [ ]: df_z= df[np.abs(fare_zscore)<=1]
        df_z
```

Out[ ]:

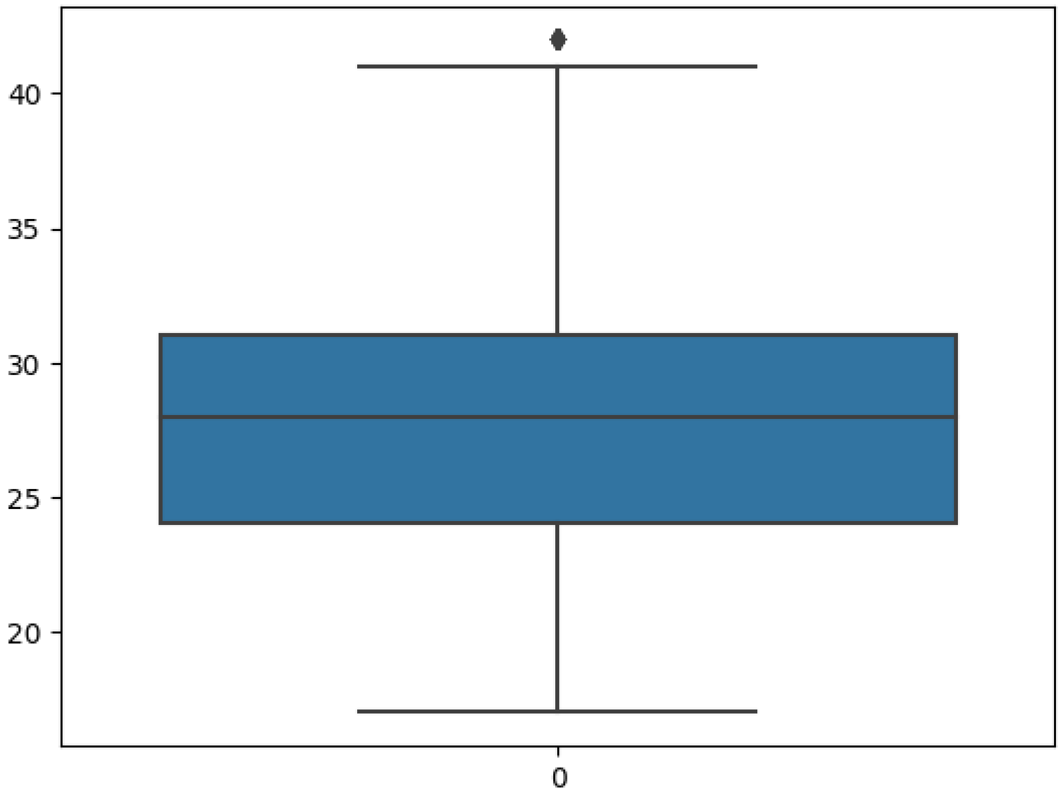
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticke
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 2117
1	2	1	1	Cumings, Mrs. John Bradley (Florence	female	38.0	1	0	PC 1759

				Briggs Th...					
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2 310128
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	11380
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	37345
...	...	...	...	...	...	...	...	...	.
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	21153
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	11205
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	28.0	1	2	W./C 660
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	11136
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	37037

662 rows × 12 columns

```
In [ ]: sns.boxplot(df_z.Age)
```

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



Percentile Method

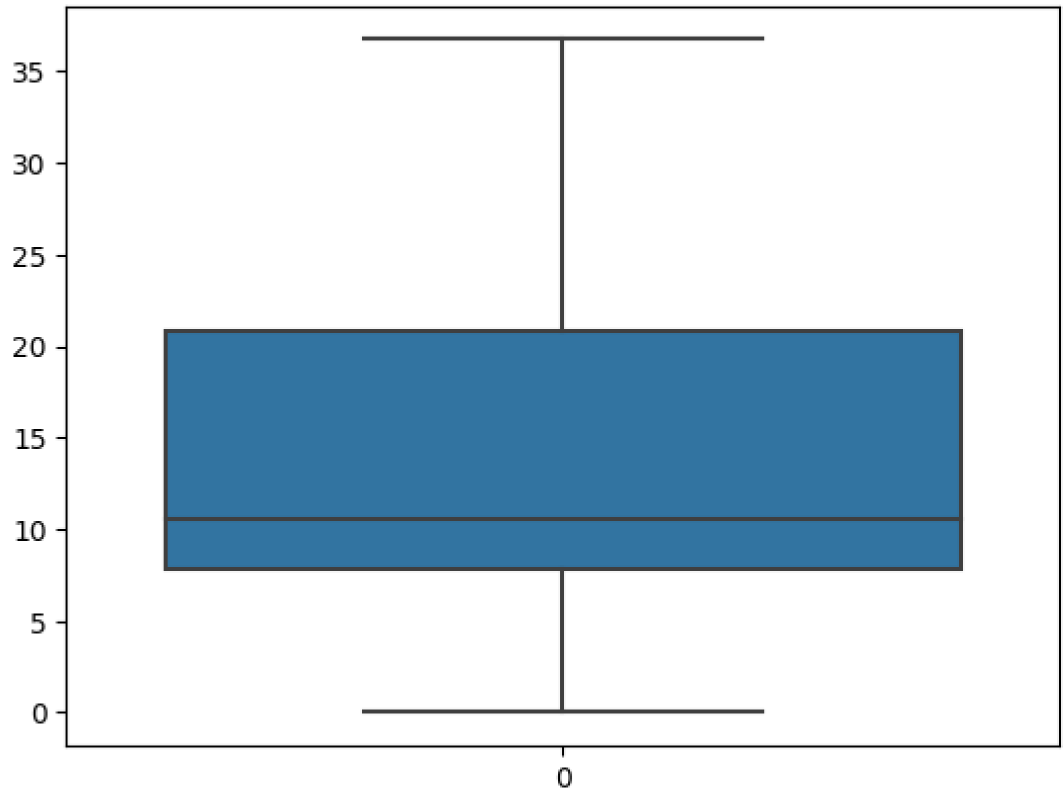
```
In [ ]: p99 = df.Fare.quantile(0.99)
        p99
```

```
Out[ ]: 36.75
```

```
In [ ]: df = df[df.Fare<=p99]
```

```
In [ ]: sns.boxplot(df.Fare)
```

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



Removal by replacement with median (For fare)

```
In [ ]: q1 = df.Fare.quantile(0.25)
        q3 = df.Fare.quantile(0.75)
```

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

```
Out[ ]: 12.933300000000003
```

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

```
Out[ ]: 40.187450000000005
```

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

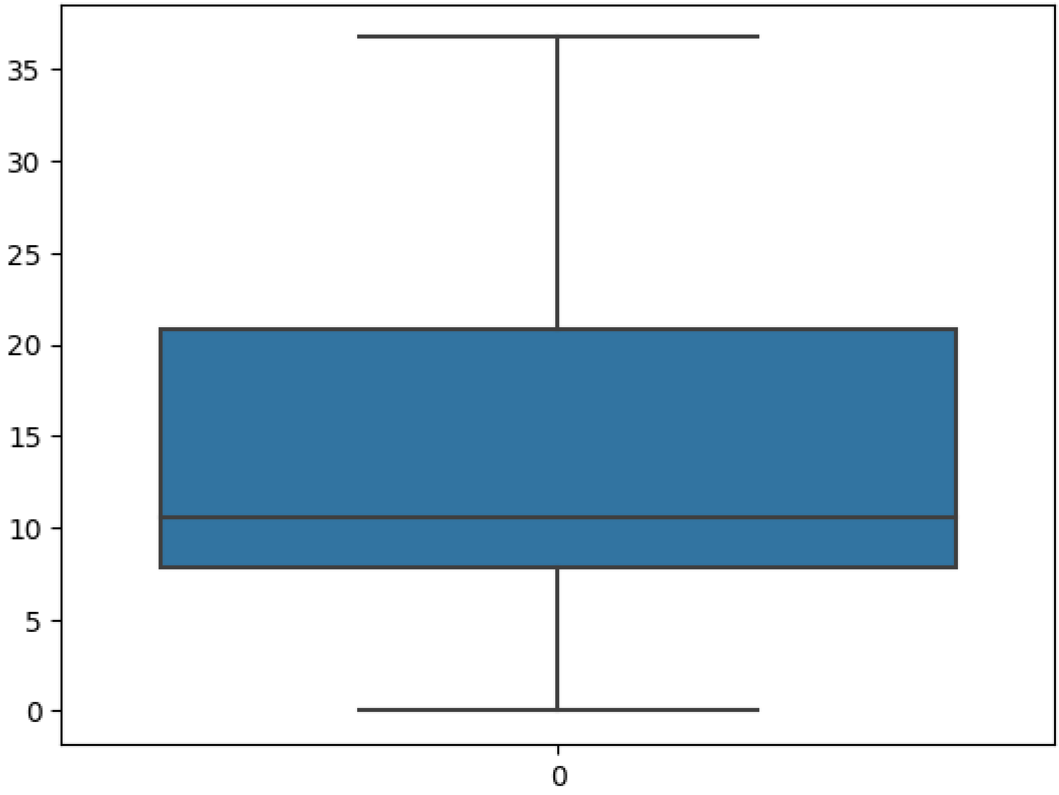
```
Out[ ]: -11.545750000000005
```

```
In [ ]: df.median(numeric_only=True)
```

```
Out[ ]: PassengerId    440.0
        Survived         0.0
        Pclass           3.0
        Age             28.0
        SibSp            0.0
        Parch            0.0
        Fare            10.5
        dtype: float64
```

```
In [ ]: df['Fare'] = np.where(df['Fare']>upper_limit,10.5,df['Fare'])
In [ ]: sns.boxplot(df['Fare'])
```

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



6. Splitting Dependent and Independent variables

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

Out[ ]:

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

In [ ]:

```
#indenpendent variables hould be 2 d array or dataframe
x= df.iloc[:,2:13]
x.head()
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Emb
0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	Unknown	
1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	Unknown	
3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	
4	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	Unknown	

In [ ]:

```
x.shape
```

Out[ ]:

```
(891, 10)
```

In [ ]:

```
type(x)
```

Out[ ]:

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

In [ ]:

```
y=df["Survived"]
y.head()
```

Out[ ]:

```
0    0
1    1
2    1
3    1
4    0
Name: Survived, dtype: int64
```

In [ ]:

```
type(y)
```

Out[ ]:

```
pandas.core.series.Series
```

7. Encoding

In [ ]:

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

In [ ]:

```
le=LabelEncoder()
```

In [ ]:

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

In [ ]:

```
x['Sex']
```

Out[ ]:

```
0      1
1      0
2      0
3      0
4      1
      ..
886    1
887    0
888    0
889    1
890    1
Name: Sex, Length: 891, dtype: int32
```

In [ ]:

```
x['Pclass'] = le.fit_transform(x['Pclass'])
x['Pclass']
```

Out[ ]:

```
0      2
1      0
2      2
3      0
4      2
      ..
886    1
887    0
888    2
889    0
890    2
Name: Pclass, Length: 891, dtype: int64
```

One hot encoding

In [ ]:

```
Embarked = pd.get_dummies(x["Embarked"])
Embarked
```

Out[ ]:

	C	Q	S
0	0	0	1
1	1	0	0
2	0	0	1
3	0	0	1
4	0	0	1
...	...	...	...
886	0	0	1
887	0	0	1
888	0	0	1
889	1	0	0
890	0	1	0

891 rows × 3 columns

In [ ]:

```
x=pd.concat([x,Embarked],axis=1)
x.head()
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embar
0	2	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	Unknown	
1	0	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	PC 17599	71.2833	C85	
2	2	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	Unknown	
3	0	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	C123	
4	2	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	Unknown	

In [ ]:

```
x.drop(["Embarked"],axis=1,inplace=True)
```

In [ ]:

```
x.head(10)
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	C	Q
0	2	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	Unknown	0	0
1	0	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	PC 17599	71.2833	C85	1	0
2	2	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	Unknown	0	0
3	0	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	C123	0	0
4	2	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	Unknown	0	0
5	2	Moran,	1	28.0	0	0	330877	8.4583	Unknown	0	1

		Mr. James									
6	0	McCarthy, Mr. Timothy J	1	54.0	0	0	17463	51.8625	E46	0	0
7	2	Palsson, Master. Gosta Leonard	1	2.0	3	1	349909	21.0750	Unknown	0	0
8	2	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	0	27.0	0	2	347742	11.1333	Unknown	0	0
9	1	Nasser, Mrs. Nicholas (Adele Achem)	0	14.0	1	0	237736	30.0708	Unknown	1	0

```
In [ ]: x.shape
```

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

8. Feature Scaling

standardiation standard scaler mean=0 and sd=1 min max scaler 0 to 1

```
In [ ]: from sklearn.preprocessing import StandardScaler
```

```
In [ ]: scaler = StandardScaler()
        x[['Age','Fare']] = scaler.fit_transform(x[['Age','Fare']])
```

```
In [ ]: x.head()
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	2	Braund, Mr. Owen Harris	1	-0.565736	1	0	A/5 21171	-0.502445	Unknown
1	0	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	0.663861	1	0	PC 17599	0.786845	C85
2	2	Heikkinen, Miss. Laina	0	-0.258337	0	0	STON/O2. 3101282	-0.488854	Unknown
3	0	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	0.433312	1	0	113803	0.420730	C123

4	2	Allen, Mr. William Henry	1	0.433312	0	0	373450	-0.486337	Unknown

9. Train Test Split

```
In [ ]: from sklearn.model_selection import train_test_split
        x_train,x_test,y_train,y_test = train_test_split(x,y,test_size =0.2,rando
```

```
In [ ]: print(x_train.shape,x_test.shape,y_train.shape,y_test.shape)
```

```
(712, 12) (179, 12) (712,) (179,)
```

```
In [ ]: x_train
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	
140	2	Boulos, Mrs. Joseph (Sultana)	0	-0.104637	0	2	2678	-0.341452	Un
439	1	Kvillner, Mr. Johan Henrik Johannesson	1	0.125912	0	0	C.A. 18723	-0.437007	Un
817	1	Mallet, Mr. Albert	1	0.125912	1	1	S.C./PARIS 2079	0.096646	Un
378	2	Betros, Mr. Tannous	1	-0.719436	0	0	2648	-0.567631	Un
491	2	Windelov, Mr. Einar	1	-0.642586	0	0	SOTON/OQ 3101317	-0.502445	Un
...	...	...	...	...	...	...	...	...	
835	0	Compton, Miss. Sara Rebecca	0	0.740711	1	1	PC 17756	1.025945	
192	2	Andersen-Jensen, Miss. Carla Christine Nielsine	0	-0.796286	1	0	350046	-0.490280	Un
629	2	O'Connell, Mr. Patrick D	1	-0.104637	0	0	334912	-0.492714	Un
559	2	de Messemaeker, Mrs. Guillaume Joseph (Emma)	0	0.510161	1	0	345572	-0.298078	Un
684	1	Brown, Mr. Thomas William Solomon	1	2.354558	1	1	29750	0.136831	Un

712 rows × 12 columns

In [ ]:
```
x_test
```

Out[ ]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabi
495	2	Youssef, Mr. Gerious	1	-0.104637	0	0	2627	-0.357308	Unknow
648	2	Willey, Mr. Edward	1	-0.104637	0	0	S.O./P.P. 751	-0.496405	Unknow
278	2	Rice, Master. Eric	1	-1.718484	4	1	382652	-0.061999	Unknow
31	0	Spencer, Mrs. William Augustus (Marie Eugenie)	0	-0.104637	1	0	PC 17569	2.301729	B7
255	2	Touma, Mrs. Darwis (Hanne Youssef Razi)	0	-0.027788	0	2	2650	-0.341452	Unknow
...	...	...	...	...	...	...	...	...	.
780	2	Ayoub, Miss. Banoura	0	-1.257385	0	0	2687	-0.502864	Unknow
837	2	Sirota, Mr. Maurice	1	-0.104637	0	0	392092	-0.486337	Unknow
215	0	Newell, Miss. Madeleine	0	0.125912	1	0	35273	1.632335	D3
833	2	Augustsson, Mr. Albert	1	-0.488887	0	0	347468	-0.490280	Unknow
372	2	Beavan, Mr. William Thomas	1	-0.796286	0	0	323951	-0.486337	Unknow

179 rows × 12 columns

In [ ]:
```
y_train
```

Out[ ]:
```
140    0
439    0
817    0
378    0
491    0
      ..
835    1
192    1
629    0
559    1
684    0
Name: Survived, Length: 712, dtype: int64
```

In [ ]:
```
y_test
```

Out[ ]:
```
495    0
648    0
```

```
278    0
31     1
255    1
      ..
780    1
837    0
215    1
833    0
372    0
Name: Survived, Length: 179, dtype: int64
```