In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

In [2]:
```
df = pd.read_csv('Titanic-Dataset.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Ca
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	I
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	I
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	I

In [4]:
```
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 [5]:
```
df.isnull().sum()
```

Out[5]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
```

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
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 [6]:
```
df = df.drop(['Cabin'],axis = 1)
```

Dropped Cabin since out of 891 values it has 687 null values

In [7]:
```
df['Age'] = df['Age'].fillna(df['Age'].median())
```

Filled all null values in age with its median

In [8]:
```
df = df.dropna()
```

In [9]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 889 entries, 0 to 890
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  889 non-null    int64  
 1   Survived     889 non-null    int64  
 2   Pclass       889 non-null    int64  
 3   Name         889 non-null    object 
 4   Sex          889 non-null    object 
 5   Age          889 non-null    float64
 6   SibSp        889 non-null    int64  
 7   Parch        889 non-null    int64  
 8   Ticket       889 non-null    object 
 9   Fare         889 non-null    float64
 10  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(4)
memory usage: 83.3+ KB
```

Dropped rows with embarked as null since there are only two rows

In [10]:
```
df.head()
```

Dropped rows with embarked as null since there are only two rows

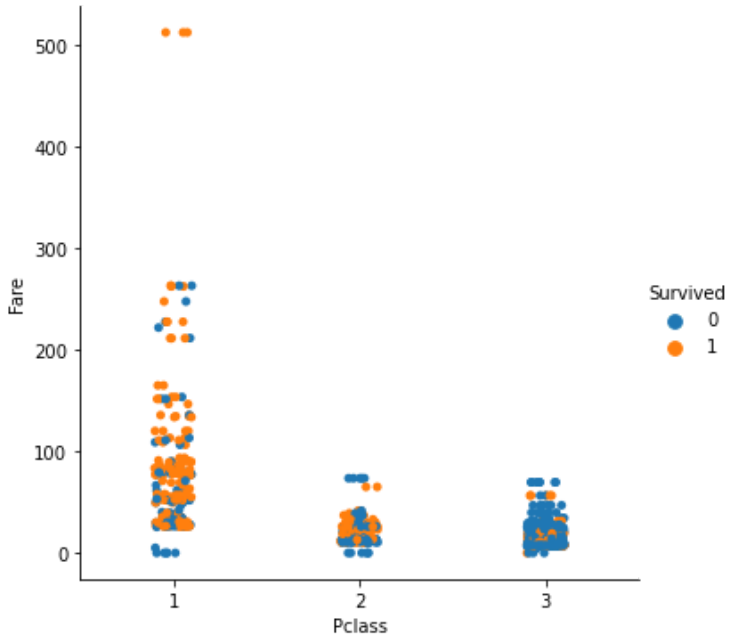
In [10]:
```
df.head()
```

Out[10]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Er
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 [11]:
```
sns.catplot(data = df,y ='Fare',x = 'Pclass',hue = 'Survived')
```

Out[11]:
```
<seaborn.axisgrid.FacetGrid at 0x20f5e4a6f10>
```



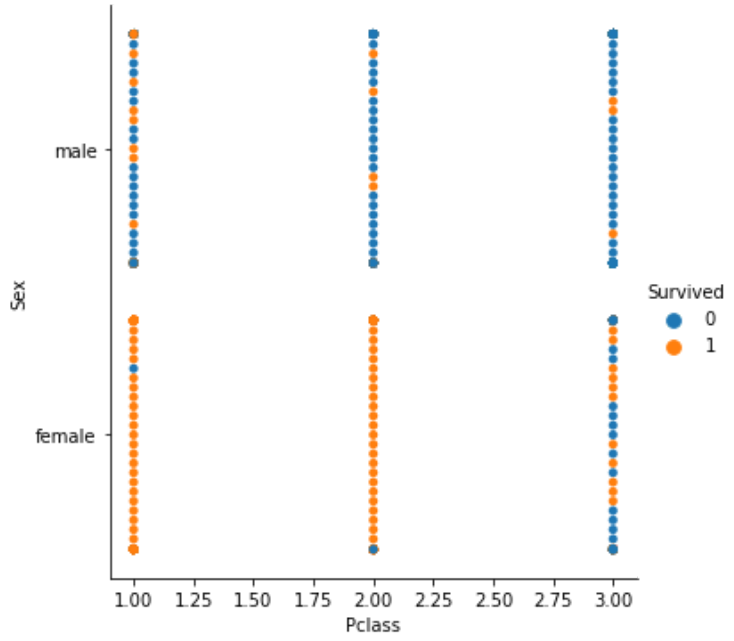
In [12]:
```
sns.catplot(data = df,y ='Sex',x = 'Pclass',hue = 'Survived',kind='swarm')
```

```
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\categorical.py:1296: User
Warning: 87.0% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\categorical.py:1296: User
```

In [12]:
```
sns.catplot(data = df,y ='Sex',x = 'Pclass',hue = 'Survived',kind='swarm')
```

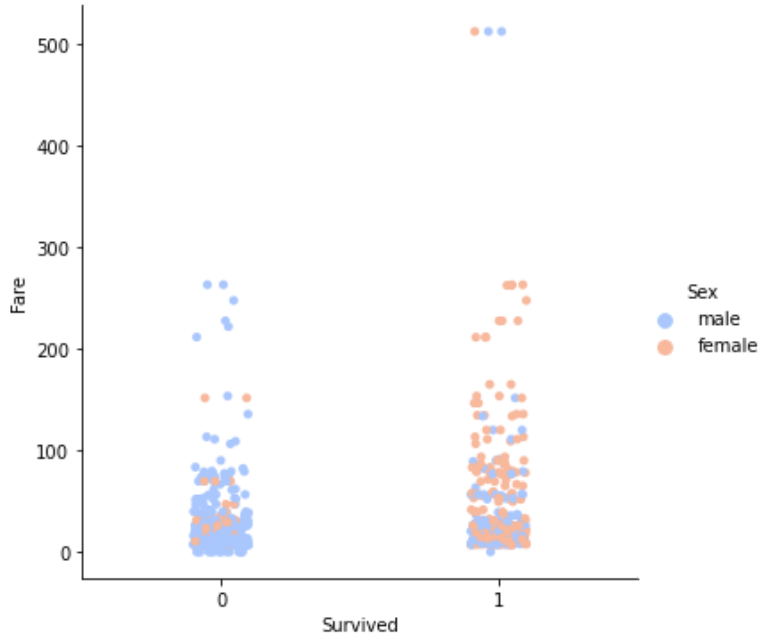
```
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\categorical.py:1296: User
Warning: 87.0% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\categorical.py:1296: User
Warning: 76.0% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
```

Out[12]: <seaborn.axisgrid.FacetGrid at 0x20f5fddaeb0>



In [13]:
```
sns.catplot(data = df, y='Fare',hue = 'Sex',x = 'Survived',palette='coolwarm')
```

Out[13]: <seaborn.axisgrid.FacetGrid at 0x20f5ff4ca00>



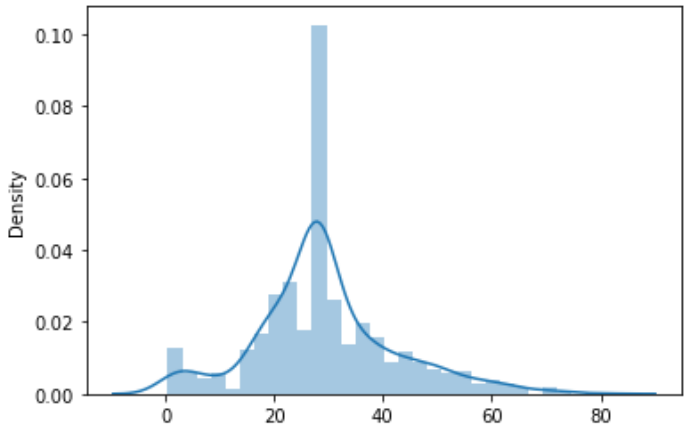
In [14]:
```
sns.distplot(x = df.sort_values(by=['Age']).Age)
```

```
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fu
tureWarning: `distplot` is a deprecated function and will be removed in a fu
ture version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `histplot` (an axes-level function for
histograms).
```

In [14]:
```
sns.distplot(x = df.sort_values(by=['Age']).Age)
```

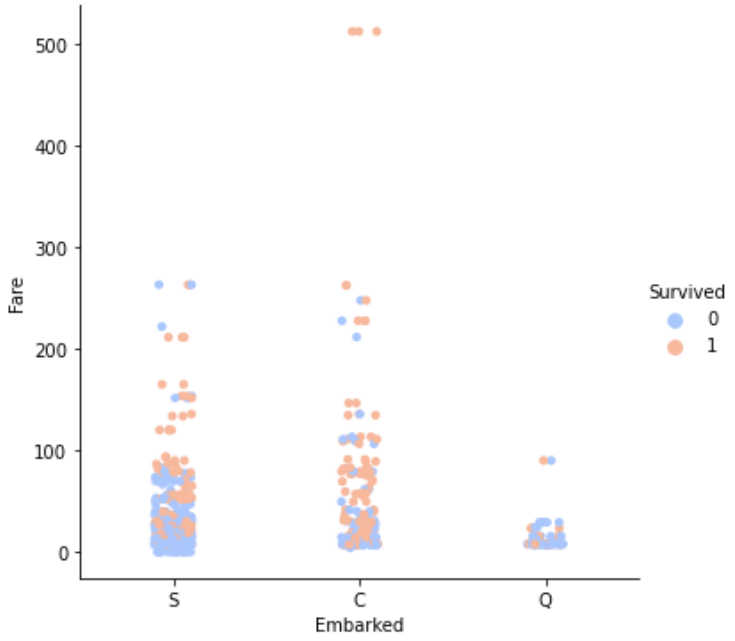
```
C:\Users\chinm\anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fu
tureWarning: `distplot` is a deprecated function and will be removed in a fu
ture version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `histplot` (an axes-level function for
histograms).
  warnings.warn(msg, FutureWarning)
```

Out[14]: <AxesSubplot:ylabel='Density'>



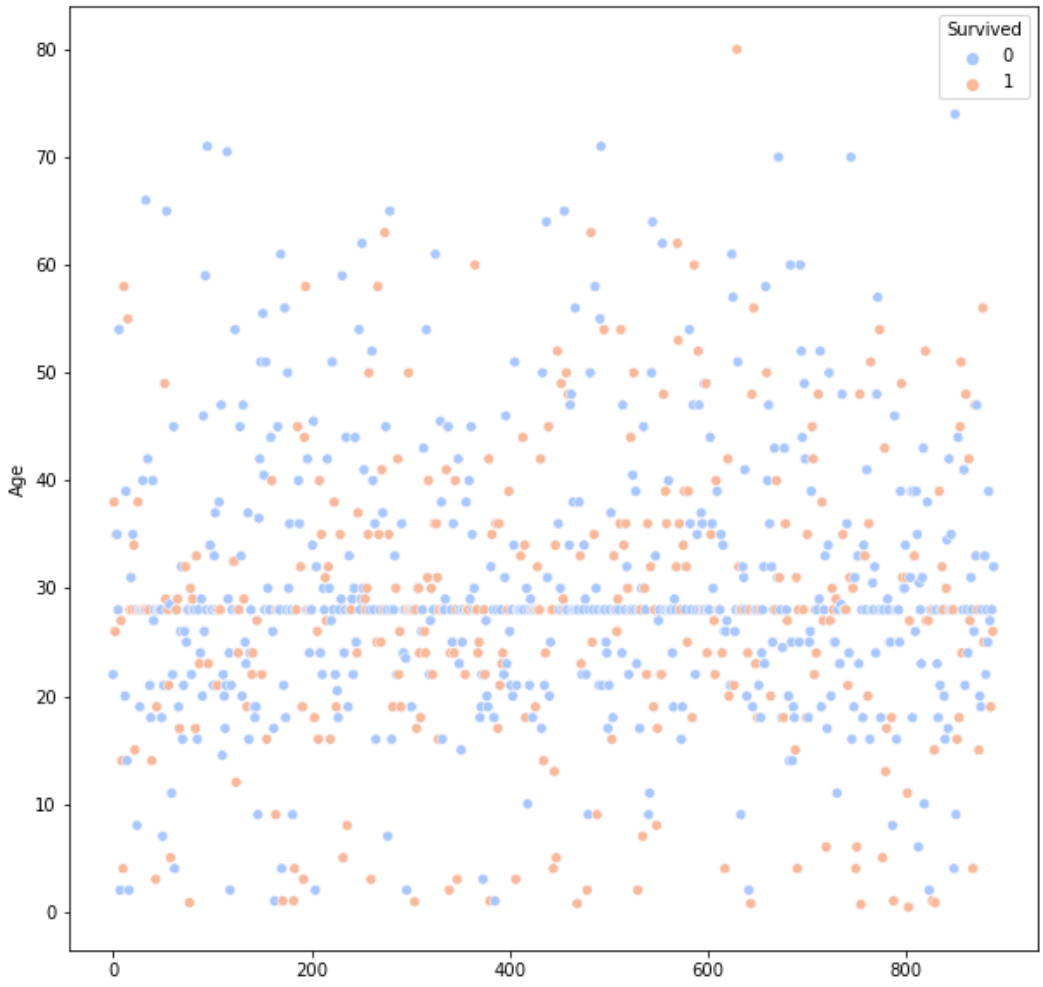
In [15]:
```
sns.catplot(x = 'Embarked', y = 'Fare', hue = 'Survived', data = df, palette =
```

Out[15]: <seaborn.axisgrid.FacetGrid at 0x20f5ff52b80>



In [16]:
```
plt.figure(figsize = (10,10))
sns.scatterplot(data = df, y = 'Age', x = np.arange(len(df['Age'])), hue = 'Su
plt.show()
```

In [16]:
```
plt.figure(figsize = (10,10))
sns.scatterplot(data = df, y = 'Age', x = np.arange(len(df['Age'])), hue = 'Su
plt.show()
```

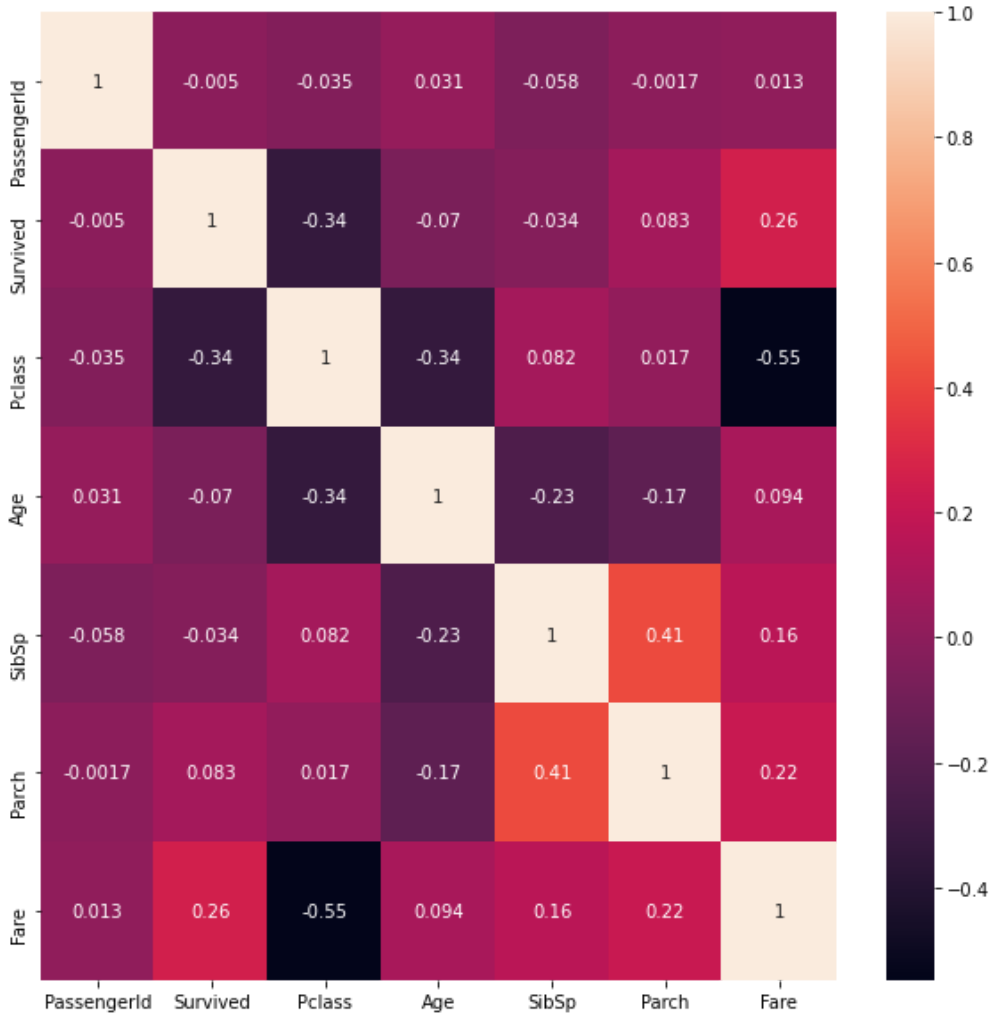


In [17]:
```
plt.figure(figsize = (10,10))
sns.heatmap(df.corr(),annot = True,cmap='rocket')
```

Out[17]: <AxesSubplot:>

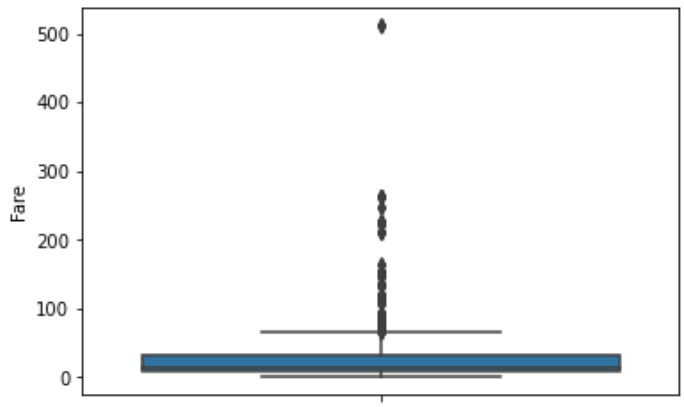
In [17]:
```
plt.figure(figsize = (10,10))
sns.heatmap(df.corr(),annot = True,cmap='rocket')
```

Out[17]: <AxesSubplot:>



In [21]:
```
sns.boxplot(data = df, y='Fare')
```

Out[21]: <AxesSubplot:ylabel='Fare'>

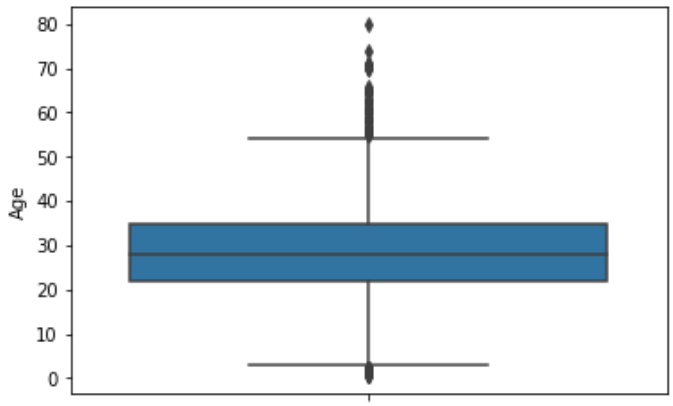


In [22]:
```
sns.boxplot(data = df, y='Age')
```

Out[22]: <AxesSubplot:ylabel='Age'>

In [22]:
```
sns.boxplot(data = df, y='Age')
```

Out[22]:
```
<AxesSubplot:ylabel='Age'>
```



The diamond shaped points symbolizes an outlier the fare column has many outliers

In [23]:
```
df.head()
```

Out[23]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Er
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 [24]:
```
x = df
```

In [ ]:
```
x = x.drop(['PassengerId','Name','Ticket'],axis = 1)
```

In [28]:
```
x
```

Out[28]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	male	22.0	1	0	7.2500	S
1	1	1	female	38.0	1	0	71.2833	C

In [28]:
```
x
```

Out[28]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	male	22.0	1	0	7.2500	S
1	1	1	female	38.0	1	0	71.2833	C
2	1	3	female	26.0	0	0	7.9250	S
3	1	1	female	35.0	1	0	53.1000	S
4	0	3	male	35.0	0	0	8.0500	S
...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S
887	1	1	female	19.0	0	0	30.0000	S
888	0	3	female	28.0	1	2	23.4500	S
889	1	1	male	26.0	0	0	30.0000	C
890	0	3	male	32.0	0	0	7.7500	Q

889 rows × 8 columns

In [29]:
```
from sklearn.preprocessing import LabelEncoder
```

In [30]:
```
le = LabelEncoder()
```

In [31]:
```
x.Sex = le.fit_transform(x.Sex)
```

In [32]:
```
x.Embarked = le.fit_transform(x.Embarked)
```

In [33]:
```
x.head()
```

Out[33]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	1	22.0	1	0	7.2500	2
1	1	1	0	38.0	1	0	71.2833	0
2	1	3	0	26.0	0	0	7.9250	2
3	1	1	0	35.0	1	0	53.1000	2
4	0	3	1	35.0	0	0	8.0500	2

In [34]:
```
from sklearn.preprocessing import MinMaxScaler
```

In [36]:
```
ms = MinMaxScaler()
```

In [37]:
```
x.columns
```

Out[37]:
```
Index(['Survived', 'Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Fare',
       'Embarked'],
      dtype='object')
```

In [38]:
```
x_scaled = pd.DataFrame(ms.fit_transform(x),columns=['Survived', 'Pclass', 'Se
```

In [39]:
```
x_scaled.head()
```

Out[39]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0.0	1.0	1.0	0.271174	0.125	0.0	0.014151	1.0
1	1.0	0.0	0.0	0.472229	0.125	0.0	0.139136	0.0

In [38]:
```
x_scaled = pd.DataFrame(ms.fit_transform(x),columns=['Survived', 'Pclass', 'Se
```

In [39]:
```
x_scaled.head()
```

Out[39]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0.0	1.0	1.0	0.271174	0.125	0.0	0.014151	1.0
1	1.0	0.0	0.0	0.472229	0.125	0.0	0.139136	0.0
2	1.0	1.0	0.0	0.321438	0.000	0.0	0.015469	1.0
3	1.0	0.0	0.0	0.434531	0.125	0.0	0.103644	1.0
4	0.0	1.0	1.0	0.434531	0.000	0.0	0.015713	1.0

In [40]:
```
from sklearn.model_selection import train_test_split
```

In [41]:
```
y = x_scaled.Survived
```

In [42]:
```
y.head()
```

Out[42]:
```
0    0.0
1    1.0
2    1.0
3    1.0
4    0.0
Name: Survived, dtype: float64
```

In [45]:
```
x_train,x_test,y_train,y_test = train_test_split(x_scaled,y,test_size= 0.3,rar
```

In [51]:
```
x_train.shape,y_train.shape,x_test.shape,y_test.shape
```

Out[51]:
```
((622, 7), (622,), (267, 7), (267,))
```