ASSIGNMENT-3

WEEK-3

NAME: N. SUSRUTHA HASINI

MORNING SESSION (10:00 – 12:00)

Perform Data preprocessing on Titanic dataset

Data Preprocessing

- Import the Libraries.
- Importing the dataset.
- Checking for Null Values.
- Data Visualization.
- Outlier Detection
- Splitting Dependent and Independent variables
- Perform Encoding
- Splitting Data into Train and Test
- Feature Scaling.

o Import the Libraries.

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

o Importing the dataset.

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	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence	female	38.0	1	0	PC 17599	71.2833	C85	C

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
				Briggs Th...								
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

In [4]:
```
df.describe()
```

Out[4]:

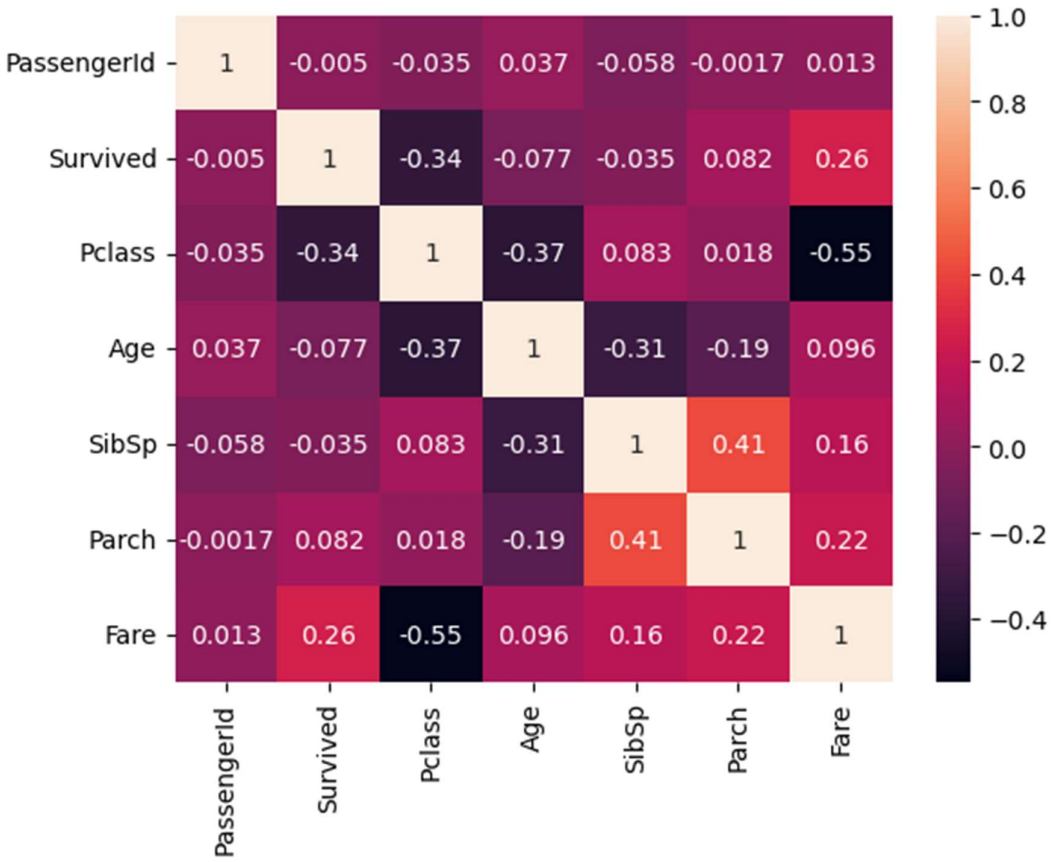
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	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

In [5]:
```
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 [6]:
```
df.corr()
```
```
/tmp/ipykernel_8747/1134722465.py:1: FutureWarning: 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 specify the value of
numeric_only to silence this warning.
  df.corr()
```
Out[6]:

	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

In [7]:
```
df.corr().Parch.sort_values(ascending=False)
```
```
/tmp/ipykernel_8747/1925834096.py:1: FutureWarning: 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 specify the value of
numeric_only to silence this warning.
  df.corr().Parch.sort_values(ascending=False)
```
Out[7]:
```
Parch          1.000000
SibSp          0.414838
Fare           0.216225
Survived       0.081629
Pclass         0.018443
PassengerId   -0.001652
Age           -0.189119
Name: Parch, dtype: float64
```
In [8]:
```
sns.heatmap(df.corr(),annot=True)
```
```
/tmp/ipykernel_8747/4277794465.py:1: FutureWarning: 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 specify the value of
numeric_only to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```
Out[8]:
```
<Axes: >
```



o Checking for null values

In [9]:
```
df.isnull().sum()
```
Out[9]:
```
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 [10]:
```
df.drop(["Cabin"],axis=1,inplace=True)
```
In [11]:
```
mean_age = df['Age'].mean()
mean_age
```
Out[11]:

```
29.69911764705882
```

In [12]:
```
df['Age'].fillna(mean_age, inplace=True)
```

In [13]:
```
mode_embarked = df["Embarked"].mode()[0]
mode_embarked
```

Out[13]:
```
'S'
```

In [14]:
```
df["Embarked"].fillna(mode_embarked,inplace=True)
```

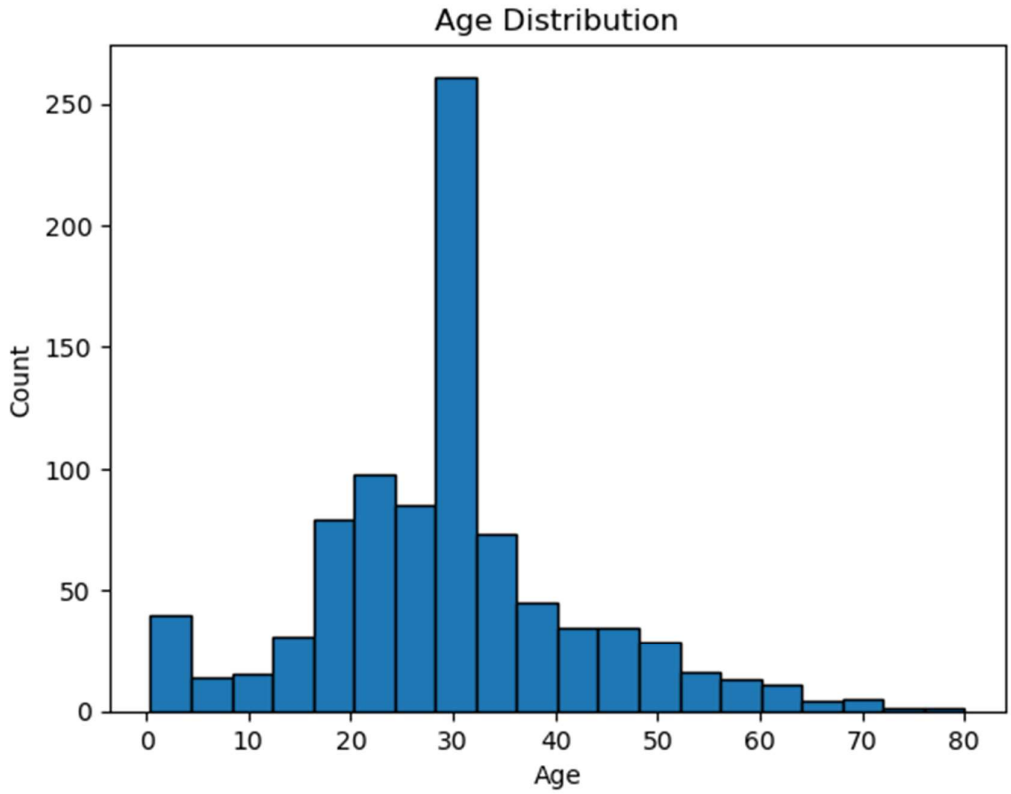
In [15]:
```
df.isnull().sum()
```

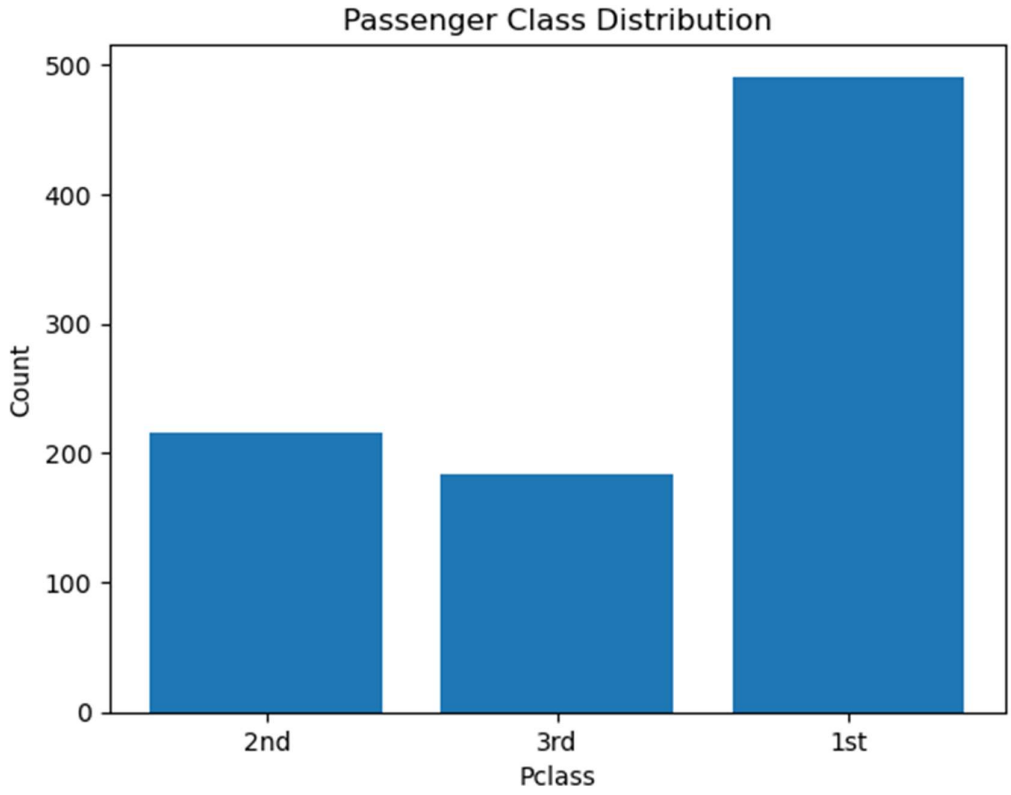
Out[15]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

o Data visualization

In [16]:
```
# Example 1: Create a histogram of the 'Age' column
plt.hist(df['Age'], bins=20, edgecolor='k')
plt.xlabel('Age')
plt.ylabel('Count')
plt.title('Age Distribution')
plt.show()

# Example 2: Create a bar chart for the 'Pclass' column
pclass_counts = df['Pclass'].value_counts()
plt.bar(pclass_counts.index, pclass_counts.values)
plt.xlabel('Pclass')
plt.ylabel('Count')
plt.title('Passenger Class Distribution')
plt.xticks(pclass_counts.index, labels=['1st', '2nd', '3rd'])
plt.show()
```



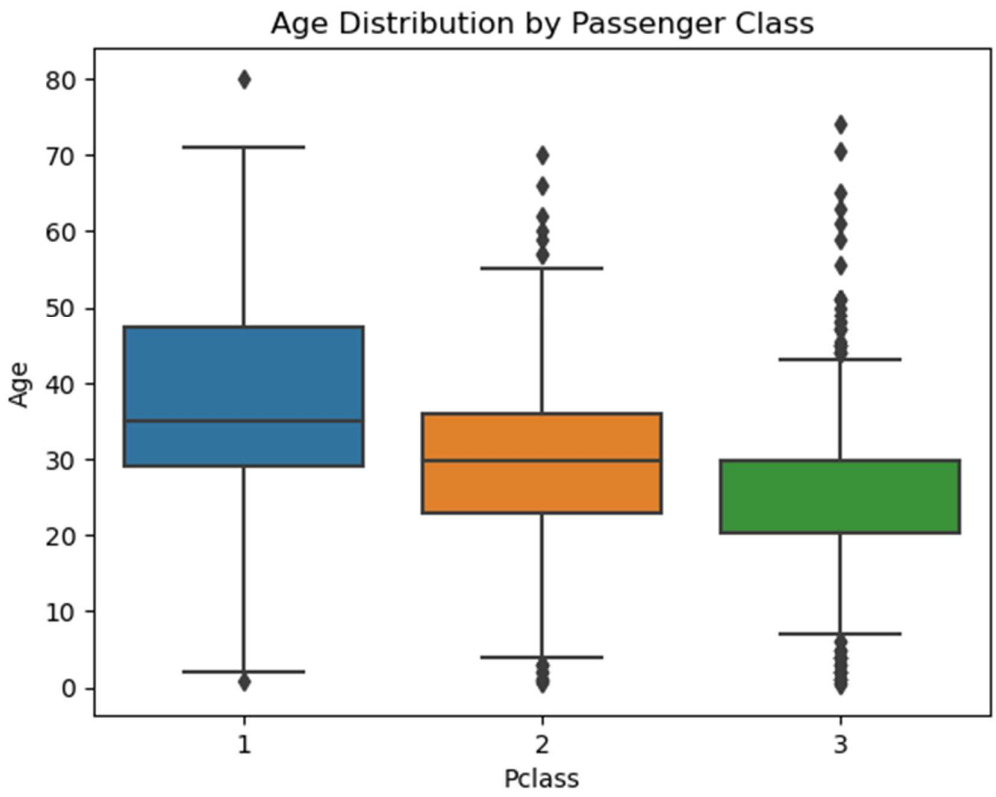


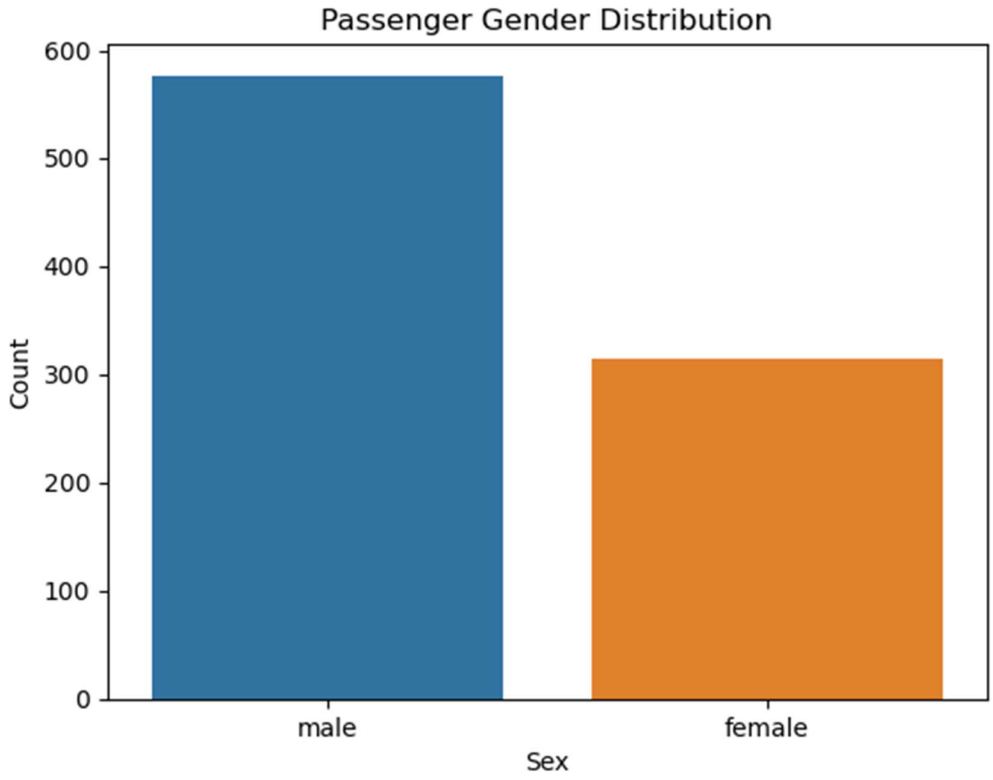
In [17]:

```
# Example 1: Create a box plot of 'Age' by 'Pclass'
sns.boxplot(x='Pclass', y='Age', data=df)
```

```
plt.xlabel('Pclass')
plt.ylabel('Age')
plt.title('Age Distribution by Passenger Class')
plt.show()

# Example 2: Create a countplot of 'Sex'
sns.countplot(x='Sex', data=df)
plt.xlabel('Sex')
plt.ylabel('Count')
plt.title('Passenger Gender Distribution')
plt.show()
```


Age Distribution by Passenger Class
Age
Pclass
0
10
20
30
40
50
60
70
80
1
2
3


Passenger Gender Distribution
Count
Sex
0
100
200
300
400
500
600
male
female

In [18]:
```
df.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 11 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          891 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  Embarked     891 non-null    object
dtypes: float64(2), int64(5), object(4)
memory usage: 76.7+ KB
```
In [19]:
```
df.drop(["PassengerId","Name","Ticket"],axis=1,inplace=True)
```
In [20]:
```
df.head()
```
Out[20]:

	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

o Outlier Detection

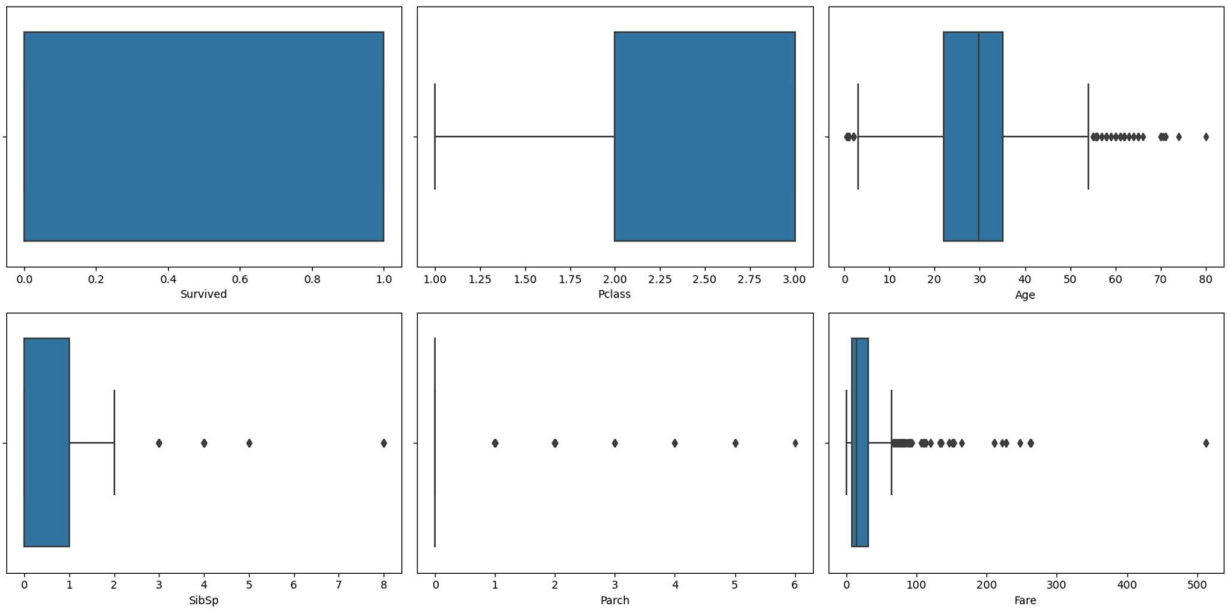
In [21]:
```
# Select numerical columns from the dataset
numerical_attributes = df.select_dtypes(include=['int64', 'float64'])

# Create box plots for each numerical attribute
plt.figure(figsize=(16, 8))  # Adjust the figure size for better
visualization

for i, column in enumerate(numerical_attributes.columns):
    plt.subplot(2, 3, i+1)  # Create subplots in a 2x3 grid
    sns.boxplot(x=column, data=numerical_attributes, whis=1.5)  # Adjust
whis as needed

plt.tight_layout()
```

```
plt.show()
```



o Removal of Outliers

In [22]:

```
# Load your dataset into a pandas DataFrame (assuming your dataset is
loaded as 'df')

# Define the numerical attributes
numerical_attributes = ['Age', 'SibSp', 'Parch', 'Fare']

# Define a function to detect and potentially remove outliers
def detect_and_remove_outliers(df, column):
    Q1 = df[column].quantile(0.25)
    Q3 = df[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = df[(df[column] < lower_bound) | (df[column] > upper_bound)]
    return outliers

# Detect and potentially remove outliers for each numerical attribute
outliers_dict = {}
for attribute in numerical_attributes:
    outliers = detect_and_remove_outliers(df, attribute)
    outliers_dict[attribute] = outliers

# Print the detected outliers for each numerical attribute
for attribute, outliers in outliers_dict.items():
    print(f"Outliers in {attribute}:")
    print(outliers)
```

```
# Optionally, remove the outliers from the DataFrame
for attribute, outliers in outliers_dict.items():
    df = df[~df.index.isin(outliers.index)]

# Now, the DataFrame 'df' contains the data with potential outliers
removed.
Outliers in Age:
     Survived  Pclass     Sex    Age  SibSp  Parch     Fare Embarked
7           0       3    male   2.00      3      1  21.0750        S
11          1       1  female  58.00      0      0  26.5500        S
15          1       2  female  55.00      0      0  16.0000        S
16          0       3    male   2.00      4      1  29.1250        Q
33          0       2    male  66.00      0      0  10.5000        S
..        ...     ...     ...    ...    ...    ...      ...      ...
827         1       2    male   1.00      0      2  37.0042        C
829         1       1  female  62.00      0      0  80.0000        S
831         1       2    male   0.83      1      1  18.7500        S
851         0       3    male  74.00      0      0   7.7750        S
879         1       1  female  56.00      0      1  83.1583        C

[66 rows x 8 columns]
Outliers in SibSp:
     Survived  Pclass     Sex        Age  SibSp  Parch      Fare Embarked
7           0       3    male   2.000000      3      1   21.0750        S
16          0       3    male   2.000000      4      1   29.1250        Q
24          0       3  female   8.000000      3      1   21.0750        S
27          0       1    male  19.000000      3      2  263.0000        S
50          0       3    male   7.000000      4      1   39.6875        S
59          0       3    male  11.000000      5      2   46.9000        S
63          0       3    male   4.000000      3      2   27.9000        S
68          1       3  female  17.000000      4      2    7.9250        S
71          0       3  female  16.000000      5      2   46.9000        S
85          1       3  female  33.000000      3      0   15.8500        S
88          1       1  female  23.000000      3      2  263.0000        S
119         0       3  female   2.000000      4      2   31.2750        S
159         0       3    male  29.699118      8      2   69.5500        S
164         0       3    male   1.000000      4      1   39.6875        S
171         0       3    male   4.000000      4      1   29.1250        Q
176         0       3    male  29.699118      3      1   25.4667        S
180         0       3  female  29.699118      8      2   69.5500        S
182         0       3    male   9.000000      4      2   31.3875        S
201         0       3    male  29.699118      8      2   69.5500        S
229         0       3  female  29.699118      3      1   25.4667        S
233         1       3  female   5.000000      4      2   31.3875        S
261         1       3    male   3.000000      4      2   31.3875        S
266         0       3    male  16.000000      4      1   39.6875        S
278         0       3    male   7.000000      4      1   29.1250        Q
324         0       3    male  29.699118      8      2   69.5500        S
341         1       1  female  24.000000      3      2  263.0000        S
```

```
374         0       3  female   3.000000      3      1   21.0750        S
386         0       3    male   1.000000      5      2   46.9000        S
409         0       3  female  29.699118      3      1   25.4667        S
480         0       3    male   9.000000      5      2   46.9000        S
485         0       3  female  29.699118      3      1   25.4667        S
541         0       3  female   9.000000      4      2   31.2750        S
542         0       3  female  11.000000      4      2   31.2750        S
634         0       3  female   9.000000      3      2   27.9000        S
642         0       3  female   2.000000      3      2   27.9000        S
683         0       3    male  14.000000      5      2   46.9000        S
686         0       3    male  14.000000      4      1   39.6875        S
726         1       2  female  30.000000      3      0   21.0000        S
787         0       3    male   8.000000      4      1   29.1250        Q
792         0       3  female  29.699118      8      2   69.5500        S
813         0       3  female   6.000000      4      2   31.2750        S
819         0       3    male  10.000000      3      2   27.9000        S
824         0       3    male   2.000000      4      1   39.6875        S
846         0       3    male  29.699118      8      2   69.5500        S
850         0       3    male   4.000000      4      2   31.2750        S
863         0       3  female  29.699118      8      2   69.5500        S
Outliers in Parch:
     Survived  Pclass     Sex        Age  SibSp  Parch     Fare Embarked
7           0       3    male   2.000000      3      1  21.0750        S
8           1       3  female  27.000000      0      2  11.1333        S
10          1       3  female   4.000000      1      1  16.7000        S
13          0       3    male  39.000000      1      5  31.2750        S
16          0       3    male   2.000000      4      1  29.1250        Q
..        ...     ...     ...        ...    ...    ...      ...      ...
871         1       1  female  47.000000      1      1  52.5542        S
879         1       1  female  56.000000      0      1  83.1583        C
880         1       2  female  25.000000      0      1  26.0000        S
885         0       3  female  39.000000      0      5  29.1250        Q
888         0       3  female  29.699118      1      2  23.4500        S

[213 rows x 8 columns]
Outliers in Fare:
     Survived  Pclass     Sex        Age  SibSp  Parch      Fare Embarked
1           1       1  female  38.000000      1      0   71.2833        C
27          0       1    male  19.000000      3      2  263.0000        S
31          1       1  female  29.699118      1      0  146.5208        C
34          0       1    male  28.000000      1      0   82.1708        C
52          1       1  female  49.000000      1      0   76.7292        C
..        ...     ...     ...        ...    ...    ...       ...      ...
846         0       3    male  29.699118      8      2   69.5500        S
849         1       1  female  29.699118      1      0   89.1042        C
856         1       1  female  45.000000      1      1  164.8667        S
863         0       3  female  29.699118      8      2   69.5500        S
879         1       1  female  56.000000      0      1   83.1583        C
[116 rows x 8 columns]
```

o Comparing the Before and After removing the outliers

In [23]:

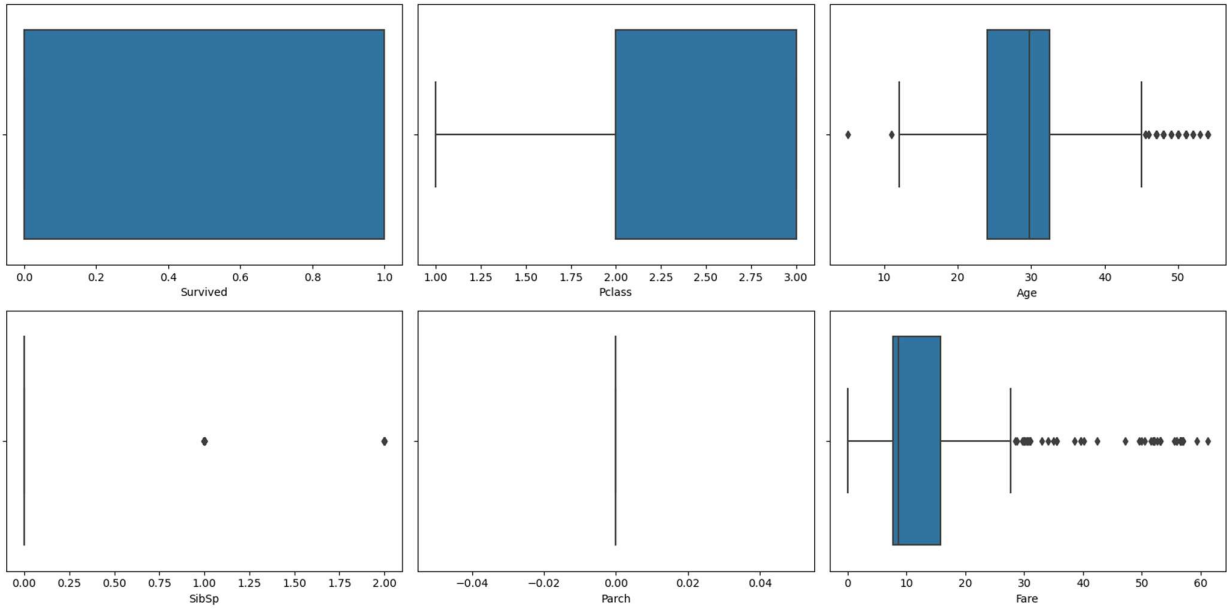
```
import matplotlib.pyplot as plt
import seaborn as sns

# Select numerical columns from the dataset
numerical_attributes = df.select_dtypes(include=['int64', 'float64'])

# Create box plots for each numerical attribute
plt.figure(figsize=(16, 8))  # Adjust the figure size for better
visualization

for i, column in enumerate(numerical_attributes.columns):
    plt.subplot(2, 3, i+1)  # Create subplots in a 2x3 grid
    sns.boxplot(x=column, data=numerical_attributes, whis=1.5)  # Adjust
whis as needed

plt.tight_layout()
plt.show()
```



In [24]:

```
df.head()
```

Out[24]:

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	male	22.000000	1	0	7.2500	S
2	1	3	female	26.000000	0	0	7.9250	S
3	1	1	female	35.000000	1	0	53.1000	S
4	0	3	male	35.000000	0	0	8.0500	S
5	0	3	male	29.699118	0	0	8.4583	Q

o Splitting the Dependent and Independent variables

In [25]:
```
# Assuming 'df' is your DataFrame containing the dataset
X = df.drop('Survived', axis=1)  # Independent variables
y = df['Survived']  # Dependent variable
```

In [26]:
```
X.head()
```

Out[26]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	male	22.000000	1	0	7.2500	S
2	3	female	26.000000	0	0	7.9250	S
3	1	female	35.000000	1	0	53.1000	S
4	3	male	35.000000	0	0	8.0500	S
5	3	male	29.699118	0	0	8.4583	Q

In [27]:
```
y.head()
```

Out[27]:
```
0    0
2    1
3    1
4    0
5    0
Name: Survived, dtype: int64
```

o Perform Encoding

In [28]:
```
from sklearn.preprocessing import LabelEncoder
```

In [29]:
```
le=LabelEncoder()
```

In [30]:
```
X["Sex"]=le.fit_transform(X["Sex"])
```

In [31]:
```
Embarked=pd.get_dummies(X["Embarked"],drop_first=True)
```

In [32]:
```
Embarked
```

Out[32]:

	Q	S
0	0	1
2	0	1
3	0	1
4	0	1
5	1	0
...	...	...
884	0	1
886	0	1

	Q	S
887	0	1
889	0	0
890	1	0

577 rows × 2 columns

In [33]:
```
X=pd.concat([X,Embarked],axis=1)
```

In [34]:
```
X.drop(["Embarked"],axis=1,inplace=True)
```

o Splitting Data into Train and Test

In [35]:
```
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,random_sta
te=2)
```

In [36]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[36]:
```
((461, 8), (116, 8), (461,), (116,))
```

o Feature Scaling

In [37]:
```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
```

In [38]:
```
x_train=sc.fit_transform(x_train)
x_test=sc.fit_transform(x_test)
```

In [39]:
```
x_train
```

Out[39]:
```
array([[ 0.6931394 , -1.76523977,  0.0155626 , ..., -0.04945919,
        -0.34879005, -1.74466606],
       [ 0.6931394 ,  0.56649528, -1.4470116 , ..., -0.5034821 ,
        -0.34879005,  0.57317559],
       [ 0.6931394 ,  0.56649528,  0.16538695, ..., -0.5612962 ,
        -0.34879005,  0.57317559],
       ...,
       [ 0.6931394 ,  0.56649528,  0.0155626 , ...,  0.7106139 ,
         2.86705424, -1.74466606],
       [-2.03794831,  0.56649528,  0.0155626 , ..., -1.18255274,
        -0.34879005,  0.57317559],
       [ 0.6931394 , -1.76523977,  0.0155626 , ..., -0.57501479,
         2.86705424, -1.74466606]])
```

In [40]:
```
x_test
```

Out[40]:
```
array([[-0.7382193 ,  0.60390884, -0.09693589, -0.39103094,  0.        ,
```

```
 -1.19986147, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589,  4.144928  ,  0.        ,
  0.75158987,  2.69920623, -1.69312335],
[-0.7382193 ,  0.60390884,  0.58763637, -0.39103094,  0.        ,
 -0.31856086, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884, -0.05807902, -0.39103094,  0.        ,
 -0.31856086, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  0.97611169, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.31636518, -0.39103094,  0.        ,
  2.75340124, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.32935021, -0.39103094,  0.        ,
 -0.40249425, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  2.3589143 , -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  1.12683927, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884,  0.84592253,  4.144928  ,  0.        ,
 -0.53468934, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  2.99680807, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.32935021, -0.39103094,  0.        ,
 -0.47278847, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884,  0.58763637, -0.39103094,  0.        ,
 -0.60183606, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884,  1.49163792, -0.39103094,  0.        ,
 -0.10872739, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.09122365, -0.39103094,  0.        ,
 -0.59134438, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -1.60779596, -0.39103094,  0.        ,
 -0.54727935, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.20020713,  1.87694853,  0.        ,
  0.13048278, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -1.73693904, -0.39103094,  0.        ,
  0.01367268, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -1.3495098 , -0.39103094,  0.        ,
 -0.40249425, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.44550826, -0.39103094,  0.        ,
  1.36010695, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.05807902, -0.39103094,  0.        ,
 -0.47278847, -0.37047929,  0.59062442],
[-2.09748023, -1.65587907,  2.91221178,  4.144928  ,  0.        ,
  3.12096234, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884, -0.83293749, -0.39103094,  0.        ,
 -0.31856086, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  1.02857006, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.20020713, -0.39103094,  0.        ,
 -0.54063183, -0.37047929,  0.59062442],
```

```
[-2.09748023,  0.60390884,  0.45849329, -0.39103094,  0.        ,
  1.02857006, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.57465133, -0.39103094,  0.        ,
 -0.53468934, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  1.31814026, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884, -0.70379441,  1.87694853,  0.        ,
  0.98240669, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.3495098 , -0.39103094,  0.        ,
 -0.54063183, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  1.36249484, -0.39103094,  0.        ,
 -0.60183606, -0.37047929,  0.59062442],
[-0.7382193 , -1.65587907,  0.58763637, -0.39103094,  0.        ,
  0.56273974, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.05807902, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[-2.09748023, -1.65587907,  2.52478254, -0.39103094,  0.        ,
  1.21007602, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -0.70379441, -0.39103094,  0.        ,
 -0.60813106, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  2.65392562, -0.39103094,  0.        ,
 -0.54937769, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.05807902, -0.39103094,  0.        ,
 -0.5930902 , -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -1.09122365, -0.39103094,  0.        ,
 -0.5930902 , -0.37047929, -1.69312335],
[ 0.62104163, -1.65587907, -1.60779596, -0.39103094,  0.        ,
 -0.57071356, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.57465133,  1.87694853,  0.        ,
  0.15146612, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  1.49163792, -0.39103094,  0.        ,
 -0.56616437, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[-2.09748023,  0.60390884,  2.13735331, -0.39103094,  0.        ,
  0.94778417, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.52419767, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589,  1.87694853,  0.        ,
  0.82712992,  2.69920623, -1.69312335],
[-0.7382193 ,  0.60390884, -0.05807902, -0.39103094,  0.        ,
 -0.10872739, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5930902 , -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -0.70379441,  1.87694853,  0.        ,
```

```
  0.29415289, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907,  2.13735331,  1.87694853,  0.        ,
  0.0171727 , -0.37047929,  0.59062442],
[-0.7382193 , -1.65587907, -0.09693589, -0.39103094,  0.        ,
  1.56994043, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.73693904, -0.39103094,  0.        ,
 -0.47278847, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.31636518, -0.39103094,  0.        ,
 -0.40249425, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.20020713, -0.39103094,  0.        ,
 -0.53468934, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  0.0171727 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  1.49163792, -0.39103094,  0.        ,
 -0.55777103, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.44550826,  1.87694853,  0.        ,
  0.01332855, -0.37047929, -1.69312335],
[-0.7382193 ,  0.60390884,  1.49163792,  1.87694853,  0.        ,
  1.06634008, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  3.54202259, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.70379441, -0.39103094,  0.        ,
 -1.19986147, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929, -1.69312335],
[-0.7382193 , -1.65587907, -0.05807902, -0.39103094,  0.        ,
 -0.10872739, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589,  1.87694853,  0.        ,
  0.01367268, -0.37047929, -1.69312335],
[-0.7382193 ,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -1.19986147, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  1.94363869, -0.39103094,  0.        ,
 -0.59344272, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.59344272, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884,  0.58763637, -0.39103094,  0.        ,
 -0.52419767, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.1872221 , -0.39103094,  0.        ,
 -0.40389594, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.1872221 , -0.39103094,  0.        ,
 -0.53888601, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884, -1.60779596, -0.39103094,  0.        ,
 -0.52419767, -0.37047929,  0.59062442],
[-0.7382193 , -1.65587907,  0.45849329, -0.39103094,  0.        ,
 -0.31856086, -0.37047929,  0.59062442],
```

```
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.1872221 , -0.39103094,  0.        ,
 -0.40249425, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.22036673, -0.39103094,  0.        ,
 -0.55077938,  2.69920623, -1.69312335],
[ 0.62104163, -1.65587907,  0.84592253, -0.39103094,  0.        ,
 -0.39515008, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884, -0.44550826, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.31636518, -0.39103094,  0.        ,
 -0.54553354, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[-2.09748023, -1.65587907,  0.58763637,  1.87694853,  0.        ,
  3.16467485, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  0.32935021, -0.39103094,  0.        ,
 -0.47348512, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -1.86608212, -0.39103094,  0.        ,
 -0.55077938,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884, -0.31636518, -0.39103094,  0.        ,
 -0.54063183, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.1872221 ,  1.87694853,  0.        ,
 -0.60848358, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.55077938,  2.69920623, -1.69312335],
[-0.7382193 , -1.65587907,  0.20020713, -0.39103094,  0.        ,
 -0.10872739, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.5930902 , -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
 -0.5371402 , -0.37047929, -1.69312335],
[-0.7382193 ,  0.60390884, -0.96208057, -0.39103094,  0.        ,
 -0.10872739, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.53853349,  2.69920623, -1.69312335],
[-0.7382193 ,  0.60390884, -1.22036673,  1.87694853,  0.        ,
 -0.23462747, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.05807902, -0.39103094,  0.        ,
 -0.59344272, -0.37047929, -1.69312335],
[ 0.62104163,  0.60390884, -1.09122365, -0.39103094,  0.        ,
 -0.60183606, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  1.40207365, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -0.09693589, -0.39103094,  0.        ,
```

```
 -0.5371402 , -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.73693904, -0.39103094,  0.        ,
 -0.60183606, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884,  1.10420868, -0.39103094,  0.        ,
  0.82712992, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.53853349,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884,  0.20020713, -0.39103094,  0.        ,
 -0.49796849, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -1.3495098 , -0.39103094,  0.        ,
 -0.47278847, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -0.09693589, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[ 0.62104163,  0.60390884,  2.26649639, -0.39103094,  0.        ,
 -0.54063183, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884, -0.09693589, -0.39103094,  0.        ,
  2.12390081, -0.37047929, -1.69312335],
[-0.7382193 , -1.65587907,  1.49163792,  1.87694853,  0.        ,
  0.98240669, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907, -1.09122365, -0.39103094,  0.        ,
 -0.37416673, -0.37047929,  0.59062442],
[-2.09748023,  0.60390884,  1.23335176, -0.39103094,  0.        ,
 -1.19986147, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.3495098 , -0.39103094,  0.        ,
 -0.52419767, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884, -1.47865288, -0.39103094,  0.        ,
 -0.31856086, -0.37047929,  0.59062442],
[ 0.62104163,  0.60390884, -1.22036673, -0.39103094,  0.        ,
 -0.54063183, -0.37047929,  0.59062442],
[ 0.62104163, -1.65587907,  0.00649252, -0.39103094,  0.        ,
 -0.54937769,  2.69920623, -1.69312335],
[-2.09748023,  0.60390884,  1.49163792,  1.87694853,  0.        ,
  3.21119074, -0.37047929,  0.59062442],
[-0.7382193 ,  0.60390884,  3.04135486, -0.39103094,  0.        ,
  0.98240669, -0.37047929,  0.59062442]])
```