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

```
import io
df = pd.read_csv('15 dataset.csv')
df.head()
```

	Ticket_ID	Age	Ticket_Price	Movie_Genre	Seat_Type	Number_of_Person	Purchase_Again
0	N4369	55.0	12.27	Comedy	Standard	7	No
1	B8091	35.0	19.02	Drama	Standard	Alone	Yes
2	V6341	55.0	22.52	Horror	VIP	3	No
3	B3243	53.0	23.01	Drama	Standard	6	Yes
4	I3814	30.0	21.81	Comedy	VIP	4	Yes

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

	0
Ticket_ID	0
Age	116
Ticket_Price	75
Movie_Genre	91
Seat_Type	0
Number_of_Person	0
Purchase_Again	0

dtype: int64

```
print(df.dropna())
```

```
     Ticket_ID         Age  Ticket_Price Movie_Genre Seat_Type  \
0        N4369   55.000000      12.27000      Comedy  Standard   
1        B8091   35.000000      19.02000       Drama  Standard   
2        V6341   55.000000      22.52000      Horror       VIP   
3        B3243   53.000000      23.01000       Drama  Standard   
4        I3814   30.000000      21.81000      Comedy       VIP   
...        ...         ...           ...         ...       ...   
1841     K4923   47.000000      12.39000      Sci-Fi  Standard   
1843     S7944  252.855055   10106.06919      Action  Standard   
1844     L3265   22.000000      15.46000       Drama       VIP   
1845     P0092   48.000000      19.63000      Action       VIP   
1846     C9487   37.000000      18.64000       Drama  Standard   

     Number_of_Person Purchase_Again  
0                   7             No  
1               Alone            Yes  
2                   3             No  
3                   6            Yes  
4                   4            Yes  
...               ...            ...  
1841            Alone            Yes  
1843            Alone            Yes  
1844            Alone            Yes  
1845            Alone            Yes  
1846                4             No  

[1582 rows x 7 columns]
```

```
df.fillna(method='ffill', inplace=True)
df.fillna(method='bfill', inplace=True)
print(df.info())

df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1847 entries, 0 to 1846
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Ticket_ID         1847 non-null   object 
 1   Age               1847 non-null   float64
 2   Ticket_Price      1847 non-null   float64
 3   Movie_Genre       1847 non-null   object 
 4   Seat_Type         1847 non-null   object 
 5   Number_of_Person  1847 non-null   object 
 6   Purchase_Again    1847 non-null   object 
dtypes: float64(2), object(5)
memory usage: 101.1+ KB
None
<ipython-input-7-8cc1d32b826a>:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Us
  df.fillna(method='ffill', inplace=True)
<ipython-input-7-8cc1d32b826a>:2: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Us
  df.fillna(method='bfill', inplace=True)
```

	Ticket_ID	Age	Ticket_Price	Movie_Genre	Seat_Type	Number_of_Person	Purchase_Again
0	N4369	55.0	12.27	Comedy	Standard	7	No
1	B8091	35.0	19.02	Drama	Standard	Alone	Yes
2	V6341	55.0	22.52	Horror	VIP	3	No
3	B3243	53.0	23.01	Drama	Standard	6	Yes
4	I3814	30.0	21.81	Comedy	VIP	4	Yes

```
df = df.drop_duplicates()
print(df)
```

```
     Ticket_ID   Age  Ticket_Price Movie_Genre Seat_Type Number_of_Person  \
0        N4369  55.0         12.27      Comedy  Standard                7   
1        B8091  35.0         19.02       Drama  Standard            Alone   
2        V6341  55.0         22.52      Horror       VIP                3   
3        B3243  53.0         23.01       Drama  Standard                6   
4        I3814  30.0         21.81      Comedy       VIP                4   
...        ...   ...           ...         ...       ...              ...   
1800     A2032  44.0         12.08      Action       VIP            Alone   
1803     G6931  27.0         11.31      Action   Premium            Alone   
1826     R8949  26.0         23.43      Comedy  Standard            Alone   
1831     A2032  44.0         16.21      Action       VIP            Alone   
1842     O7454  47.0         16.34      Comedy   Premium            Alone   

     Purchase_Again  
0                No  
1               Yes  
2                No  
3               Yes  
4               Yes  
...             ...  
1800            Yes  
1803             No  
1826            Yes  
1831            Yes  
1842            Yes  

[1522 rows x 7 columns]
```

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

```
<ipython-input-11-fd76c88bd32c>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained ass
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting 

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


  df["Age"].fillna(df["Age"].mode(), inplace=True)
<ipython-input-11-fd76c88bd32c>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained ass
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting 

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


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

```
df
```

	Ticket_ID	Age	Ticket_Price	Movie_Genre	Seat_Type	Number_of_Person	Purchase_Again
0	N4369	55.0	12.27	Comedy	Standard	7	No
1	B8091	35.0	19.02	Drama	Standard	Alone	Yes
2	V6341	55.0	22.52	Horror	VIP	3	No
3	B3243	53.0	23.01	Drama	Standard	6	Yes
4	I3814	30.0	21.81	Comedy	VIP	4	Yes
...	...	...	...	...	...	...	...
1800	A2032	44.0	12.08	Action	VIP	Alone	Yes
1803	G6931	27.0	11.31	Action	Premium	Alone	No
1826	R8949	26.0	23.43	Comedy	Standard	Alone	Yes
1831	A2032	44.0	16.21	Action	VIP	Alone	Yes
1842	O7454	47.0	16.34	Comedy	Premium	Alone	Yes

1522 rows × 7 columns

```
df['Movie_Genre'].fillna(df['Movie_Genre'].mode()[0], inplace=True)
```

```
<ipython-input-14-cdcc0498e1f4>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained ass
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


  df['Movie_Genre'].fillna(df['Movie_Genre'].mode()[0], inplace=True)
```

```
df
```

	Ticket_ID	Age	Ticket_Price	Movie_Genre	Seat_Type	Number_of_Person	Purchase_Again
0	N4369	55.0	12.27	Comedy	Standard	7	No
1	B8091	35.0	19.02	Drama	Standard	Alone	Yes
2	V6341	55.0	22.52	Horror	VIP	3	No
3	B3243	53.0	23.01	Drama	Standard	6	Yes
4	I3814	30.0	21.81	Comedy	VIP	4	Yes
...	...	...	...	...	...	...	...
1800	A2032	44.0	12.08	Action	VIP	Alone	Yes
1803	G6931	27.0	11.31	Action	Premium	Alone	No
1826	R8949	26.0	23.43	Comedy	Standard	Alone	Yes
1831	A2032	44.0	16.21	Action	VIP	Alone	Yes
1842	O7454	47.0	16.34	Comedy	Premium	Alone	Yes

1522 rows × 7 columns

```
Q1 = df['Ticket_Price'].quantile(0.25)
Q3 = df['Ticket_Price'].quantile(0.75)
IQR = Q3 - Q1

# Define bounds
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

outliers = df[(df['Ticket_Price'] < lower_bound) | (df['Ticket_Price'] > upper_bound)]
print("\nOutliers detected:\n", outliers)
```

```
Outliers detected:
      Ticket_ID         Age  Ticket_Price Movie_Genre Seat_Type  \
8        A2029  252.855055   10106.06919      Sci-Fi  Standard
20       F5223  252.855055   10106.06919      Horror  Standard
31       E7092  252.855055   10106.06919      Comedy   Premium
37       A6397  252.855055   10106.06919       Drama   Premium
40       I6521  252.855055   10106.06919      Comedy       VIP
...        ...         ...           ...         ...       ...
1457     I2406  252.855055   10000.00000      Sci-Fi   Premium
1466     U7953  252.855055   10106.06919      Comedy   Premium
1481     R4165  200.000000   10000.00000      Horror       VIP
1482     S9225  200.000000   10000.00000      Sci-Fi  Standard
```

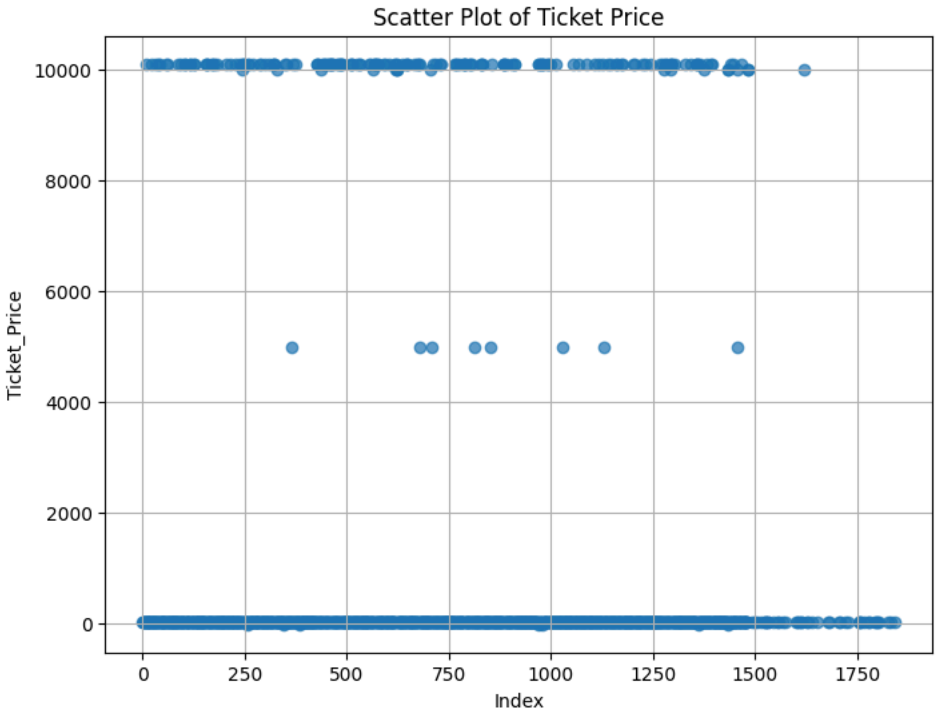
```
1619     R4165   21.000000   10000.00000      Horror       VIP

     Number_of_Person Purchase_Again
8               Alone            Yes
20              Alone             No
31                  7             No
37                  5            Yes
40                  7             No
...               ...            ...
1457                3            Yes
1466                3            Yes
1481                4             No
1482            Alone            Yes
1619                4             No

[194 rows x 7 columns]
```

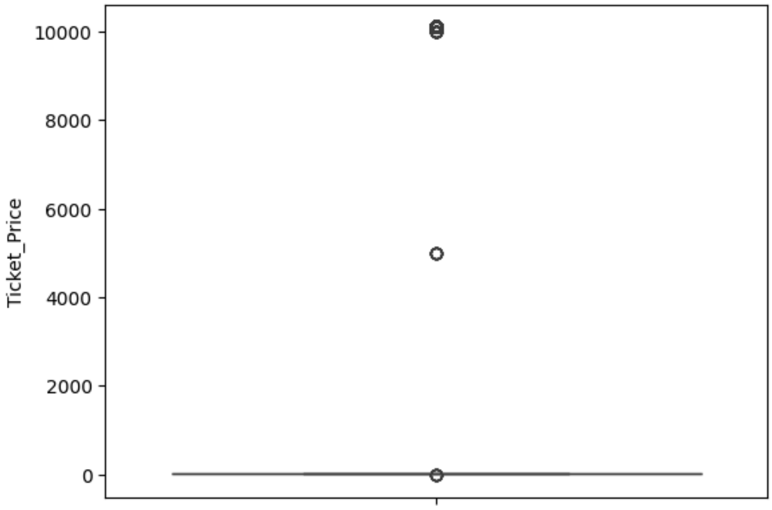
Double-click (or enter) to edit

```
plt.figure(figsize=(8, 6))
plt.scatter(df.index, df['Ticket_Price'], marker='o', alpha=0.7)
plt.xlabel('Index')
plt.ylabel('Ticket_Price')
plt.title('Scatter Plot of Ticket Price')
plt.grid(True)
plt.show()
```



```
sns.boxplot(data=df, y='Ticket_Price')
```

```
<Axes: ylabel='Ticket_Price'>
```



```
df.to_csv('cleaned_dataset.csv', index=False)
```

```
Start coding or generate with AI.
```