```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

# load data

```
df = pd.read_csv("Amazon Sales data.csv",parse_dates = ['Order
Date','Ship Date'])

df.head()
```

```
                               Region                 Country
Item Type  \
0              Australia and Oceania                  Tuvalu
Baby Food
1  Central America and the Caribbean                 Grenada
Cereal
2                              Europe                  Russia  Office
Supplies
3                  Sub-Saharan Africa  Sao Tome and Principe
Fruits
4                  Sub-Saharan Africa                 Rwanda  Office
Supplies

  Sales Channel Order Priority Order Date   Order ID  Ship Date  Units
Sold  \
0       Offline              H 2010-05-28  669165933 2010-06-27
9925
1        Online              C 2012-08-22  963881480 2012-09-15
2804
2       Offline              L 2014-05-02  341417157 2014-05-08
1779
3        Online              C 2014-06-20  514321792 2014-07-05
8102
4       Offline              L 2013-02-01  115456712 2013-02-06
5062

   Unit Price  Unit Cost  Total Revenue  Total Cost  Total Profit
0      255.28     159.42     2533654.00  1582243.50     951410.50
1      205.70     117.11      576782.80   328376.44     248406.36
2      651.21     524.96     1158502.59   933903.84     224598.75
3        9.33       6.92       75591.66    56065.84      19525.82
4      651.21     524.96     3296425.02  2657347.52     639077.50
```

```
df.shape
```

```
(100, 14)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 14 columns):
 #   Column          Non-Null Count  Dtype         
---  ------          --------------  -----         
 0   Region          100 non-null    object        
 1   Country         100 non-null    object        
 2   Item Type       100 non-null    object        
 3   Sales Channel   100 non-null    object        
 4   Order Priority  100 non-null    object        
 5   Order Date      100 non-null    datetime64[ns]
 6   Order ID        100 non-null    int64         
 7   Ship Date       100 non-null    datetime64[ns]
 8   Units Sold      100 non-null    int64         
 9   Unit Price      100 non-null    float64       
 10  Unit Cost       100 non-null    float64       
 11  Total Revenue   100 non-null    float64       
 12  Total Cost      100 non-null    float64       
 13  Total Profit    100 non-null    float64       
dtypes: datetime64[ns](2), float64(5), int64(2), object(5)
memory usage: 11.1+ KB
```

# Finding Missing Values

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

```
Region            0
Country           0
Item Type         0
Sales Channel     0
Order Priority    0
Order Date        0
Order ID          0
Ship Date         0
Units Sold        0
Unit Price        0
Unit Cost         0
Total Revenue     0
Total Cost        0
Total Profit      0
dtype: int64
```

# Finding and removing duplicate rows

```
df.duplicated().sum()
```

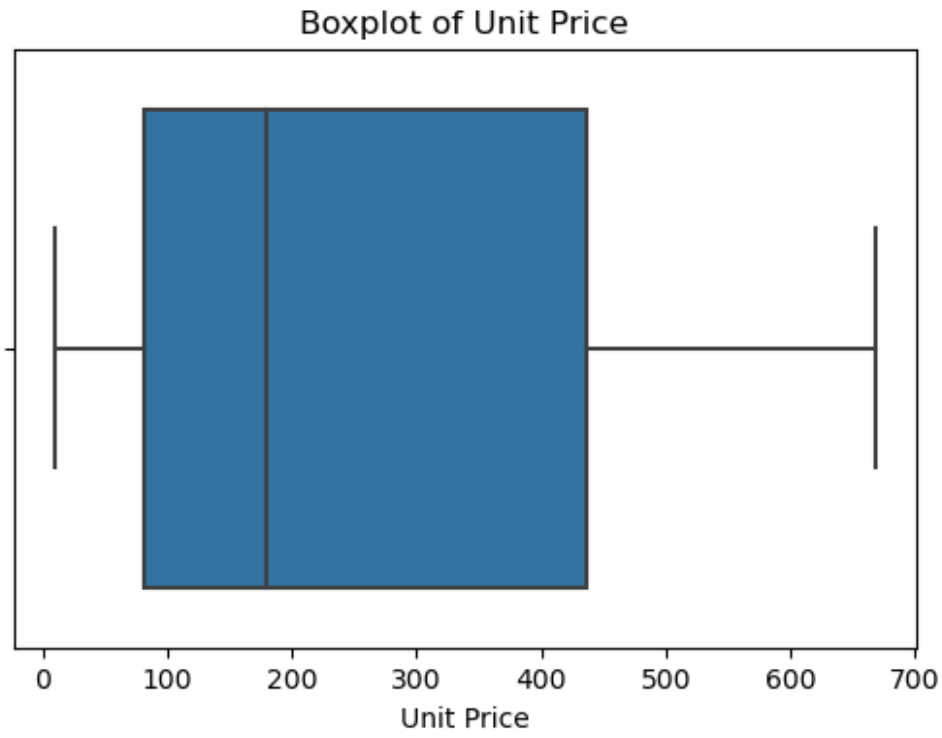
```
0
```

```
df.describe()
```

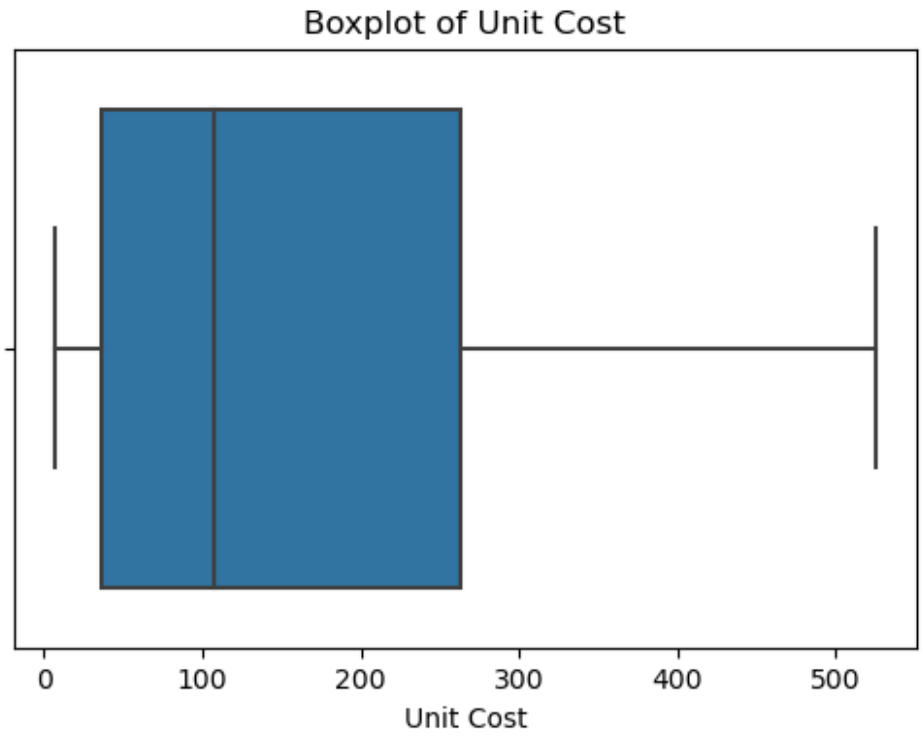
```
           Order ID   Units Sold  Unit Price   Unit Cost  Total
Revenue  \
count  1.000000e+02   100.000000  100.000000  100.000000
1.000000e+02
mean   5.550204e+08  5128.710000  276.761300  191.048000
1.373488e+06
std    2.606153e+08  2794.484562  235.592241  188.208181
1.460029e+06
min    1.146066e+08   124.000000    9.330000    6.920000
4.870260e+03
25%    3.389225e+08  2836.250000   81.730000   35.840000
2.687212e+05
50%    5.577086e+08  5382.500000  179.880000  107.275000
7.523144e+05
75%    7.907551e+08  7369.000000  437.200000  263.330000
2.212045e+06
max    9.940222e+08  9925.000000  668.270000  524.960000
5.997055e+06

         Total Cost  Total Profit
count  1.000000e+02  1.000000e+02
mean   9.318057e+05  4.416820e+05
std    1.083938e+06  4.385379e+05
min    3.612240e+03  1.258020e+03
25%    1.688680e+05  1.214436e+05
50%    3.635664e+05  2.907680e+05
75%    1.613870e+06  6.358288e+05
max    4.509794e+06  1.719922e+06
```

# Finding Outliers

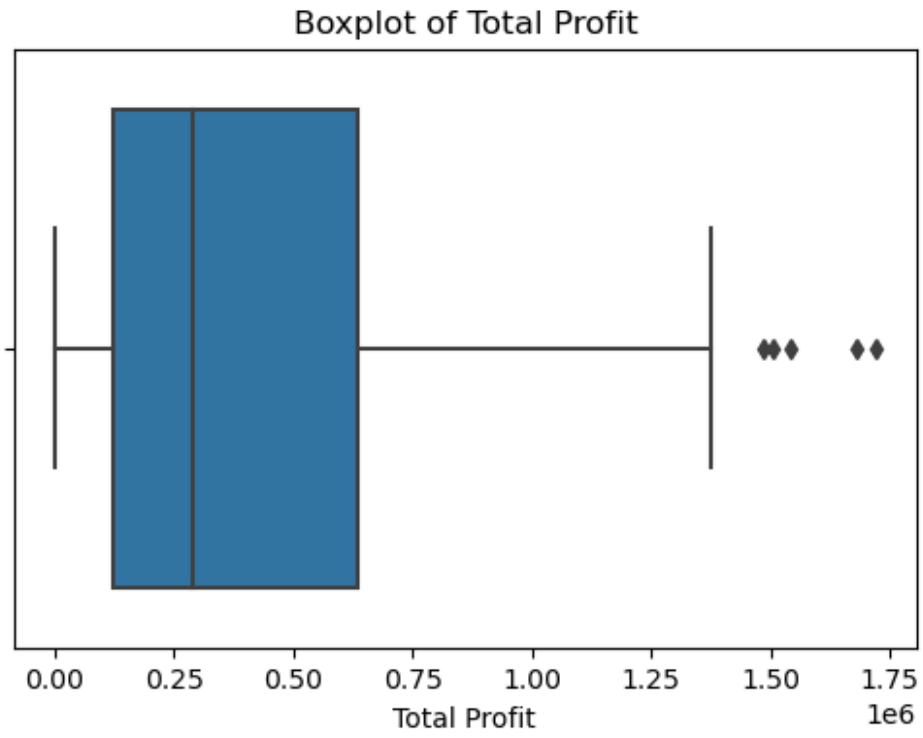
```
plt.figure(figsize=(6, 4))
sns.boxplot(x=df['Unit Price'])
plt.title('Boxplot of Unit Price')
plt.show()
```




```
plt.figure(figsize=(6, 4))
sns.boxplot(x=df['Unit Cost'])
plt.title('Boxplot of Unit Cost')
plt.show()
```




```
plt.figure(figsize=(6, 4))
sns.boxplot(x=df['Total Profit'])
plt.title('Boxplot of Total Profit')
plt.show()
```




# IQR method to remove outliers

Observation: total Profit contains outliers with the help of IQR technique Outliers will be removed

```
Q1 = df['Total Profit'].quantile(0.25)
Q3 = df['Total Profit'].quantile(0.70)

IQR = Q3 - Q1

lower_limit = Q1 - 1.5 * IQR
lower_limit
```

```
-561653.7735000001
```

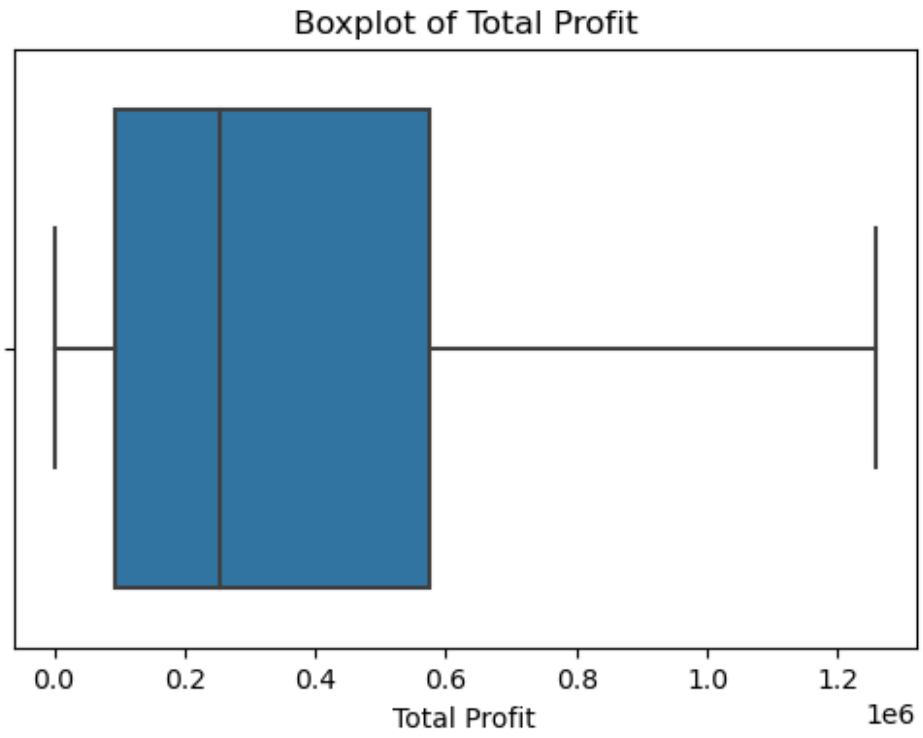
```
upper_limit = Q3 + 1.5 * IQR
upper_limit
```

```
1259939.1825
```

```
df[(df['Total Profit'] < lower_limit) | (df['Total Profit'] >
upper_limit)]
```

```
                                Region      Country  Item Type Sales
Channel  \
13  Central America and the Caribbean     Honduras  Household
```

```
Offline
30                              Europe  Switzerland  Cosmetics
Offline
33                                Asia      Myanmar  Household
Offline
46                              Europe      Iceland  Cosmetics
Online
74       Middle East and North Africa     Pakistan  Cosmetics
Offline
79               Australia and Oceania        Samoa   Cosmetics
Online
93                              Europe      Romania  Cosmetics
Online

   Order Priority Order Date   Order ID  Ship Date  Units Sold  Unit
Price  \
13              H 2017-02-08  522840487 2017-02-13        8974
668.27
30              M 2012-09-17  249693334 2012-10-20        8661
437.20
33              H 2015-01-16  177713572 2015-03-01        8250
668.27
46              C 2016-12-31  331438481 2016-12-31        8867
437.20
74              L 2013-07-05  231145322 2013-08-16        9892
437.20
79              H 2013-07-20  670854651 2013-08-07        9654
437.20
93              H 2010-11-26  660643374 2010-12-25        7910
437.20

    Unit Cost  Total Revenue  Total Cost  Total Profit
13     502.54     5997054.98  4509793.96    1487261.02
30     263.33     3786589.20  2280701.13    1505888.07
33     502.54     5513227.50  4145955.00    1367272.50
46     263.33     3876652.40  2334947.11    1541705.29
74     263.33     4324782.40  2604860.36    1719922.04
79     263.33     4220728.80  2542187.82    1678540.98
93     263.33     3458252.00  2082940.30    1375311.70

df2= df[(df['Total Profit'] > lower_limit) & (df['Total Profit'] <
upper_limit)]

plt.figure(figsize=(6, 4))
sns.boxplot(x=df2['Total Profit'])
plt.title('Boxplot of Total Profit')
plt.show()
```




```
for col in df2.describe(include = 'object').columns:
    print(col)
    print(df2[col].unique())
    print('-'*50)
```

```
Region
['Australia and Oceania' 'Central America and the Caribbean' 'Europe'
 'Sub-Saharan Africa' 'Asia' 'Middle East and North Africa'
 'North America']
--------------------------------------------------
Country
['Tuvalu' 'Grenada' 'Russia' 'Sao Tome and Principe' 'Rwanda'
 'Solomon Islands' 'Angola' 'Burkina Faso' 'Republic of the Congo'
 'Senegal' 'Kyrgyzstan' 'Cape Verde' 'Bangladesh' 'Mongolia'
'Bulgaria'
 'Sri Lanka' 'Cameroon' 'Turkmenistan' 'East Timor' 'Norway'
'Portugal'
 'Honduras' 'New Zealand' 'Moldova ' 'France' 'Kiribati' 'Mali'
 'The Gambia' 'South Sudan' 'Australia' 'Djibouti' 'Costa Rica'
'Syria'
 'Brunei' 'Niger' 'Azerbaijan' 'Slovakia' 'Myanmar' 'Comoros'
 'Switzerland' 'Macedonia' 'Mauritania' 'Albania' 'Lesotho' 'Saudi
Arabia'
 'Sierra Leone' "Cote d'Ivoire" 'Fiji' 'Austria' 'United Kingdom'
 'San Marino' 'Libya' 'Haiti' 'Gabon' 'Belize' 'Lithuania'
'Madagascar'
 'Democratic Republic of the Congo' 'Mexico'
```

```
 'Federated States of Micronesia' 'Laos' 'Monaco' 'Spain' 'Lebanon'
'Iran'
 'Zambia' 'Kenya' 'Kuwait' 'Slovenia' 'Nicaragua' 'Malaysia'
'Mozambique']
--------------------------------------------------
Item Type
['Baby Food' 'Cereal' 'Office Supplies' 'Fruits' 'Household'
'Vegetables'
 'Personal Care' 'Clothes' 'Cosmetics' 'Beverages' 'Meat' 'Snacks']
--------------------------------------------------
Sales Channel
['Offline' 'Online']
--------------------------------------------------
Order Priority
['H' 'C' 'L' 'M']
--------------------------------------------------
```

```
new_data=df2.to_csv("new amazon1.csv")
```

```
df2.shape
```

```
(93, 14)
```