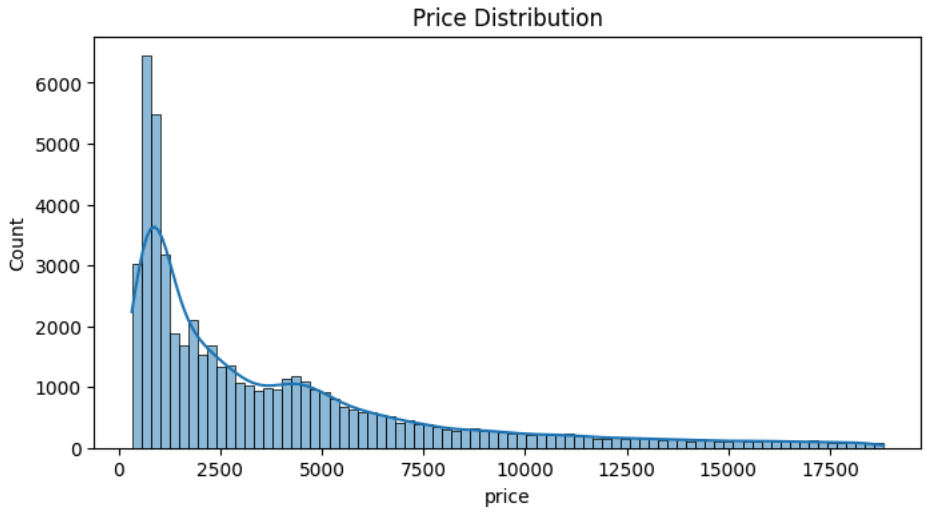
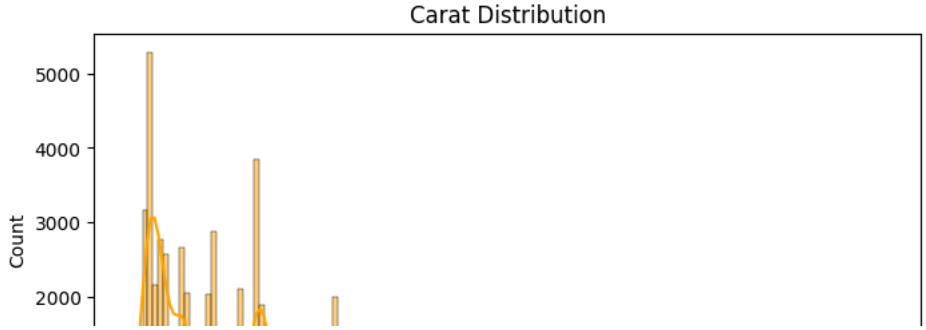
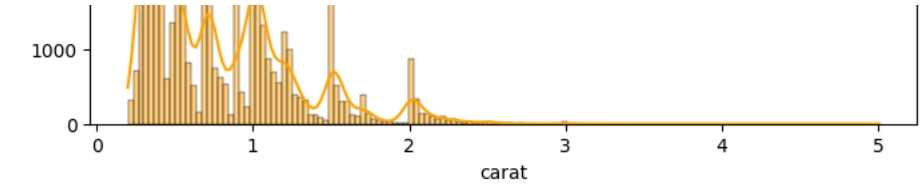
```
# 1 Import required libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# 2 Load the dataset
df = sns.load_dataset('diamonds')

# 3 View first few rows
print(df.head())

# 4 Check dataset information
df.info()

# 5 Basic statistical summary
print(df.describe())

# 6 Check for missing values
print(df.isnull().sum())

# 7 Univariate Analysis

# Price distribution
plt.figure(figsize=(8, 4))
sns.histplot(df['price'], kde=True)
plt.title("Price Distribution")
plt.show()

# Carat distribution
plt.figure(figsize=(8, 4))
sns.histplot(df['carat'], kde=True, color='orange')
plt.title("Carat Distribution")
plt.show()

# 8 Bivariate Analysis

# Scatter plot for Carat vs Price
plt.figure(figsize=(8, 4))
sns.scatterplot(x='carat', y='price', data=df)
plt.title("Carat vs Price")
plt.show()

# Boxplot for Price vs Cut
plt.figure(figsize=(8, 4))
sns.boxplot(x='cut', y='price', data=df)
plt.title("Price vs Cut")
plt.show()

# Boxplot for Price vs Color
plt.figure(figsize=(8, 4))
sns.boxplot(x='color', y='price', data=df)
plt.title("Price vs Color")
plt.show()

# Boxplot for Price vs Clarity
plt.figure(figsize=(8, 4))
sns.boxplot(x='clarity', y='price', data=df)
plt.title("Price vs Clarity")
plt.show()

# 9 Correlation Analysis

# Select only numeric columns for correlation to avoid error
numeric_df = df.select_dtypes(include=['float64', 'int64'])

# Heatmap for correlation
plt.figure(figsize=(8, 6))
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()

# 10 Outlier Detection using boxplot for 'price'
plt.figure(figsize=(8, 4))
sns.boxplot(x=df['price'])
plt.title("Outliers in Price")
plt.show()

# 11 Outlier Detection using boxplot for 'carat'
plt.figure(figsize=(8, 4))
sns.boxplot(x=df['carat'])
```

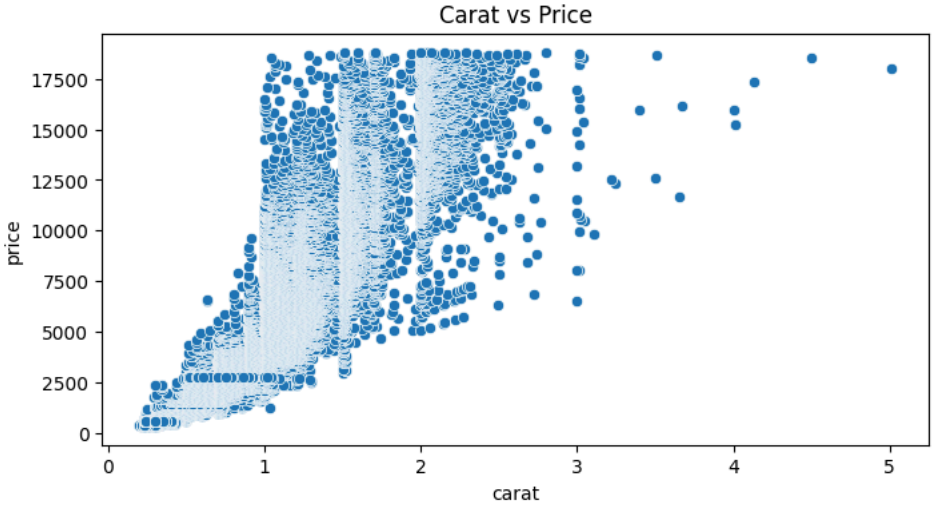
```
    plt.title("Outliers in Carat")
    plt.show()
```

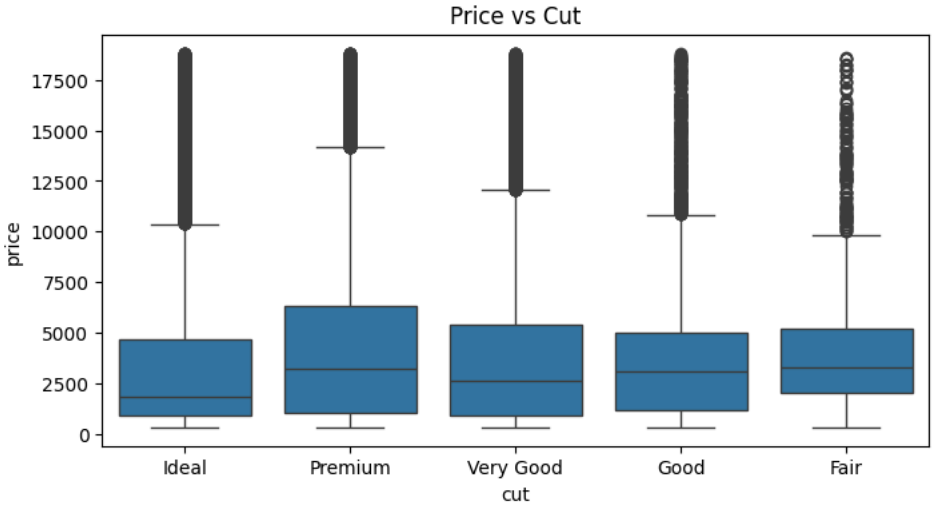
```
   carat      cut color clarity  depth  table  price     x     y     z
0   0.23    Ideal     E     SI2   61.5   55.0    326  3.95  3.98  2.43
1   0.21  Premium     E     SI1   59.8   61.0    326  3.89  3.84  2.31
2   0.23     Good     E     VS1   56.9   65.0    327  4.05  4.07  2.31
3   0.29  Premium     I     VS2   62.4   58.0    334  4.20  4.23  2.63
4   0.31     Good     J     SI2   63.3   58.0    335  4.34  4.35  2.75
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 53940 entries, 0 to 53939
Data columns (total 10 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   carat    53940 non-null  float64
 1   cut      53940 non-null  category
 2   color    53940 non-null  category
 3   clarity  53940 non-null  category
 4   depth    53940 non-null  float64
 5   table    53940 non-null  float64
 6   price    53940 non-null  int64
 7   x        53940 non-null  float64
 8   y        53940 non-null  float64
 9   z        53940 non-null  float64
dtypes: category(3), float64(6), int64(1)
memory usage: 3.0 MB
              carat         depth         table         price             x  \
count  53940.000000  53940.000000  53940.000000  53940.000000  53940.000000
mean       0.797940     61.749405     57.457184   3932.799722      5.731157
std        0.474011      1.432621      2.234491   3989.439738      1.121761
min        0.200000     43.000000     43.000000    326.000000      0.000000
25%        0.400000     61.000000     56.000000    950.000000      4.710000
50%        0.700000     61.800000     57.000000   2401.000000      5.700000
75%        1.040000     62.500000     59.000000   5324.250000      6.540000
max        5.010000     79.000000     95.000000  18823.000000     10.740000

                 y             z
count  53940.000000  53940.000000
mean       5.734526      3.538734
std        1.142135      0.705699
min        0.000000      0.000000
25%        4.720000      2.910000
50%        5.710000      3.530000
75%        6.540000      4.040000
max       58.900000     31.800000
carat      0
cut        0
color      0
clarity    0
depth      0
table      0
price      0
x          0
y          0
z          0
dtype: int64
```

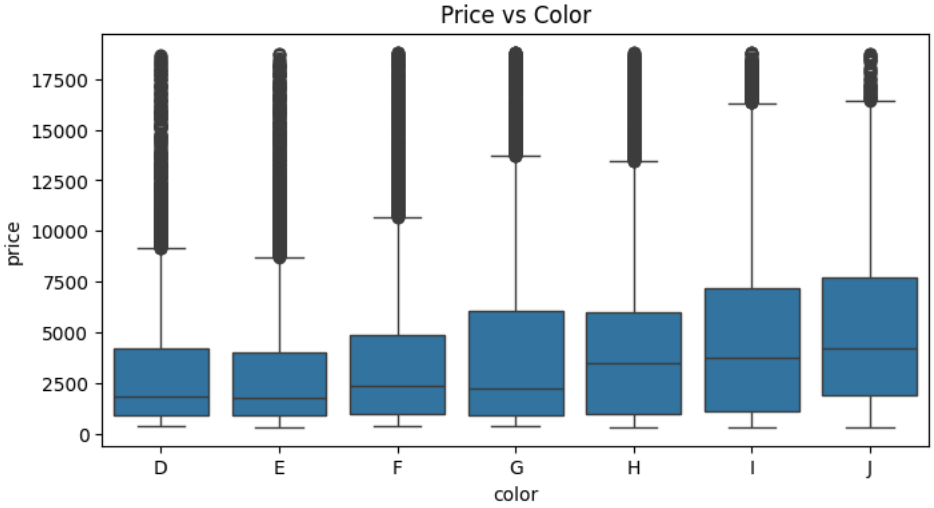


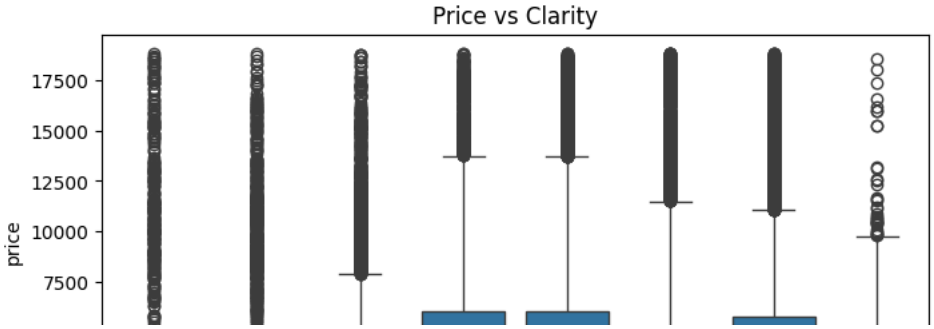


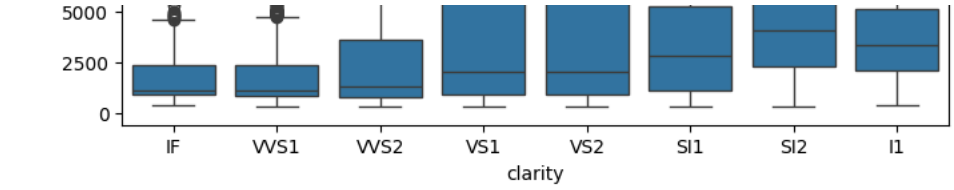

1000
0
0
1
2
3
4
5
carat

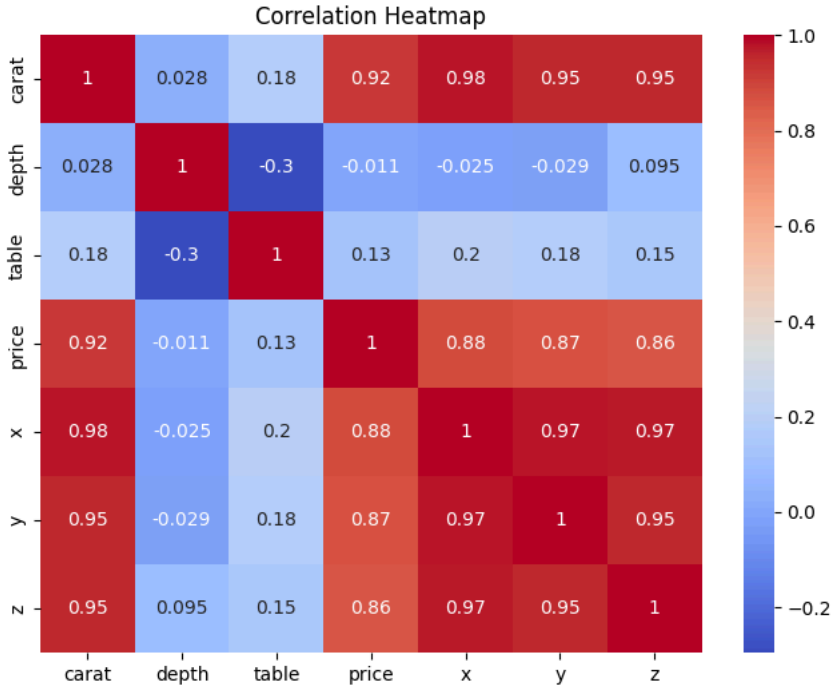

Carat vs Price
price
17500
15000
12500
10000
7500
5000
2500
0
0
1
2
3
4
5
carat

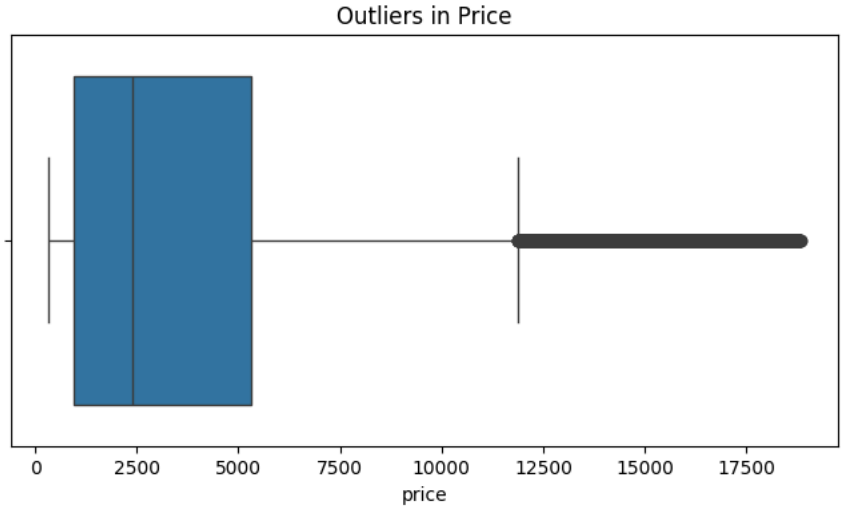

Price vs Cut
price
17500
15000
12500
10000
7500
5000
2500
0
Ideal
Premium
Very Good
Good
Fair
cut

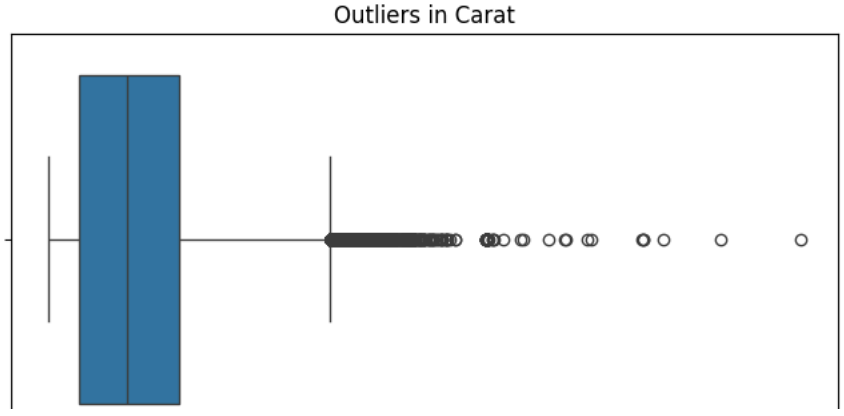

Price vs Color
price
17500
15000
12500
10000
7500
5000
2500
0
D
E
F
G
H
I
J
color


Price vs Clarity
price
17500
15000
12500
10000
7500


5000
2500
0
IF
VVS1
VVS2
VS1
VS2
SI1
SI2
I1
clarity


Correlation Heatmap
carat
depth
table
price
x
y
z
1
0.028
0.18
0.92
0.98
0.95
0.95
0.028
1
-0.3
-0.011
-0.025
-0.029
0.095
0.18
-0.3
1
0.13
0.2
0.18
0.15
0.92
-0.011
0.13
1
0.88
0.87
0.86
0.98
-0.025
0.2
0.88
1
0.97
0.97
0.95
-0.029
0.18
0.87
0.97
1
0.95
0.95
0.095
0.15
0.86
0.97
0.95
1
1.0
0.8
0.6
0.4
0.2
0.0
−0.2


Outliers in Price
0
2500
5000
7500
10000
12500
15000
17500
price


Outliers in Carat