nm-assignment3

October 28, 2023

[ ]:

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

[ ]:

```
supermarket_sales_data = pd.read_csv("House Price India.csv")
```

[ ]:

```
supermarket_sales_data.head()
```

[ ]:

```
           id   Date  number of bedrooms  number of bathrooms  living area  \
0  6762810145  42491                   5                 2.50         3650
1  6762810635  42491                   4                 2.50         2920
2  6762810998  42491                   5                 2.75         2910
3  6762812605  42491                   4                 2.50         3310
4  6762812919  42491                   3                 2.00         2710

   lot area  number of floors  waterfront present  number of views  \
0      9050               2.0                   0                4
1      4000               1.5                   0                0
2      9480               1.5                   0                0
3     42998               2.0                   0                0
4      4500               1.5                   0                0

   condition of the house  …  Built Year  Renovation Year  Postal Code  \
0                       5  …        1921                0       122003
1                       5  …        1909                0       122004
2                       3  …        1939                0       122004
3                       3  …        2001                0       122005
4                       4  …        1929                0       122006

   Lattitude  Longitude  living_area_renov  lot_area_renov  \
0    52.8645   -114.557               2880            5400
1    52.8878   -114.470               2470            4000
2    52.8852   -114.468               2940            6600
3    52.9532   -114.321               3350           42847
4    52.9047   -114.485               2060            4500
```

```
   Number of schools nearby  Distance from the airport    Price
0                         2                         58  2380000
1                         2                         51  1400000
2                         1                         53  1200000
3                         3                         76   838000
4                         1                         51   805000

[5 rows x 23 columns]
```

[ ]:

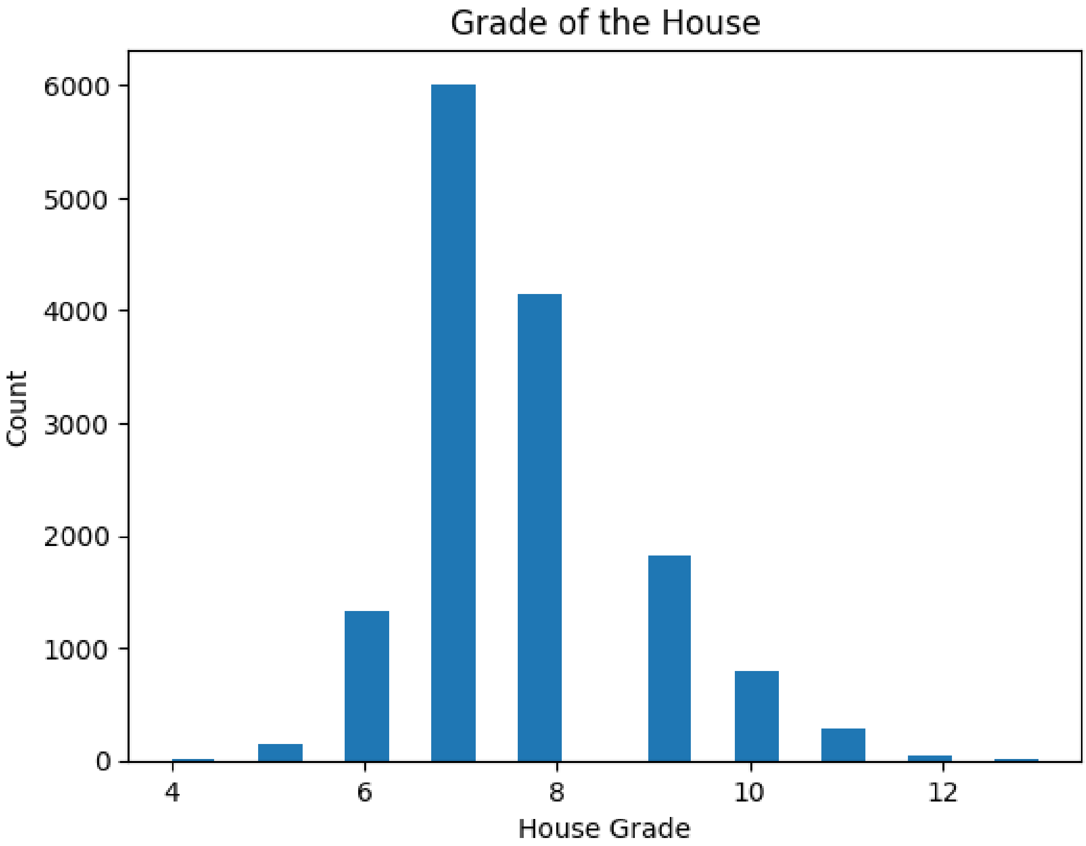
```
# returns a dataframe with column names of the dataset
pd.DataFrame(list(supermarket_sales_data.columns), columns=['Column Name'])
```

[ ]:

```
                                  Column Name
0                                          id
1                                        Date
2                          number of bedrooms
3                         number of bathrooms
4                                 living area
5                                    lot area
6                            number of floors
7                          waterfront present
8                             number of views
9                      condition of the house
10                         grade of the house
11   Area of the house(excluding basement)
12                      Area of the basement
13                                 Built Year
14                           Renovation Year
15                                Postal Code
16                                  Lattitude
17                                  Longitude
18                          living_area_renov
19                             lot_area_renov
20                   Number of schools nearby
21                  Distance from the airport
22                                      Price
```

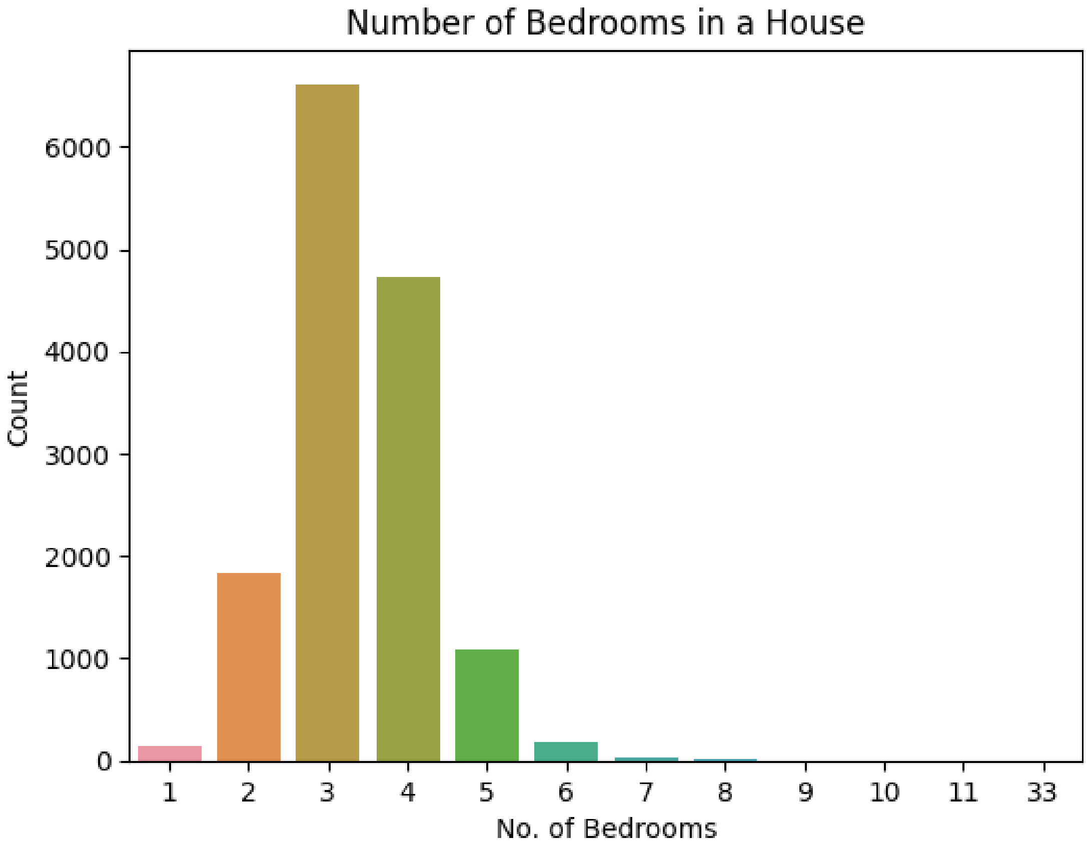
[ ]:

```
## UNIVARIATE ANALYSIS ##
# Histogram
supermarket_sales_data['grade of the house'].plot.hist(bins=20)
plt.xlabel('House Grade')
plt.ylabel('Count')
plt.title('Grade of the House')
plt.show()
```

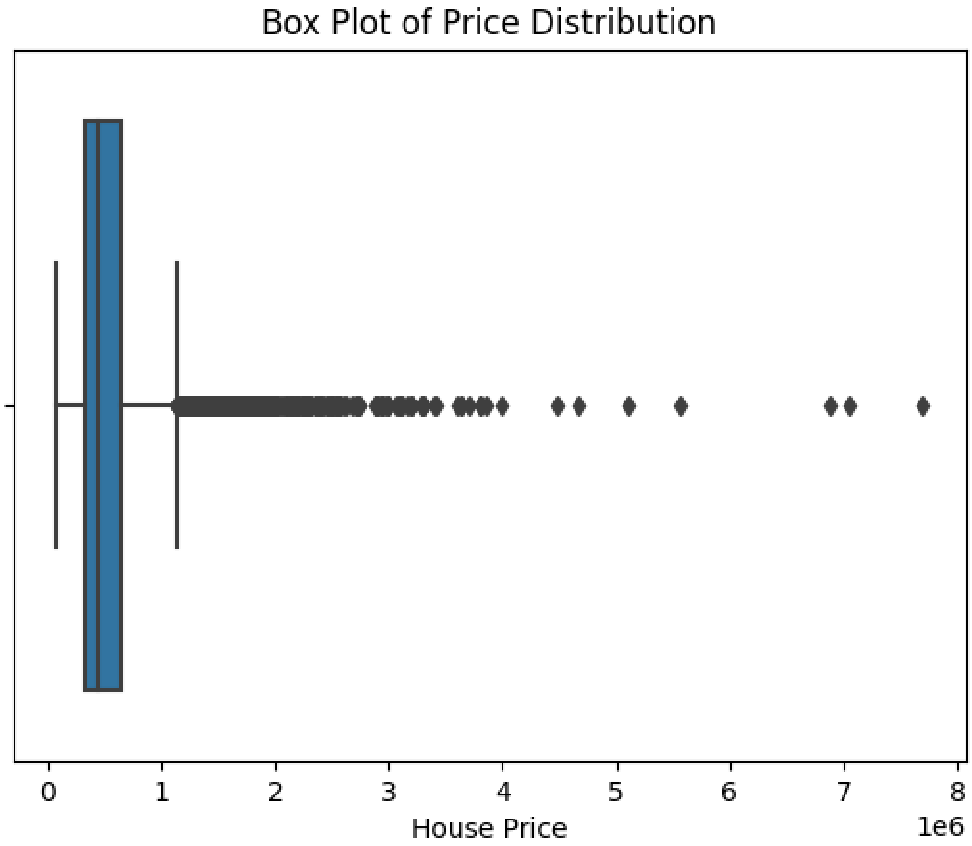


[ ]:

```
# returns the countplot of bedrooms
sns.countplot(data=supermarket_sales_data, x='number of bedrooms')
plt.xlabel('No. of Bedrooms')
plt.ylabel('Count')
plt.title('Number of Bedrooms in a House')
plt.xticks(rotation=0)
plt.show()
```

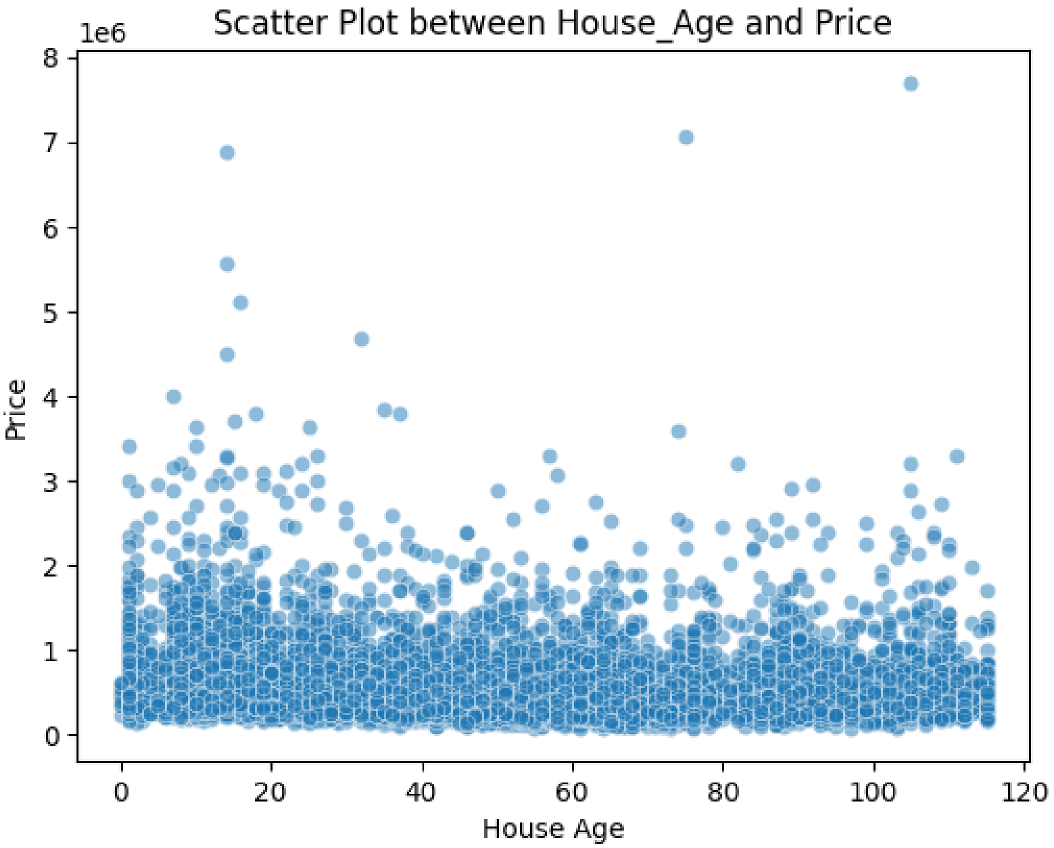


[ ]:
```
# Box Plot for Outliers
sns.boxplot(data=supermarket_sales_data, x='Price')
plt.xlabel('House Price')
plt.title('Box Plot of Price Distribution')
plt.show()
```



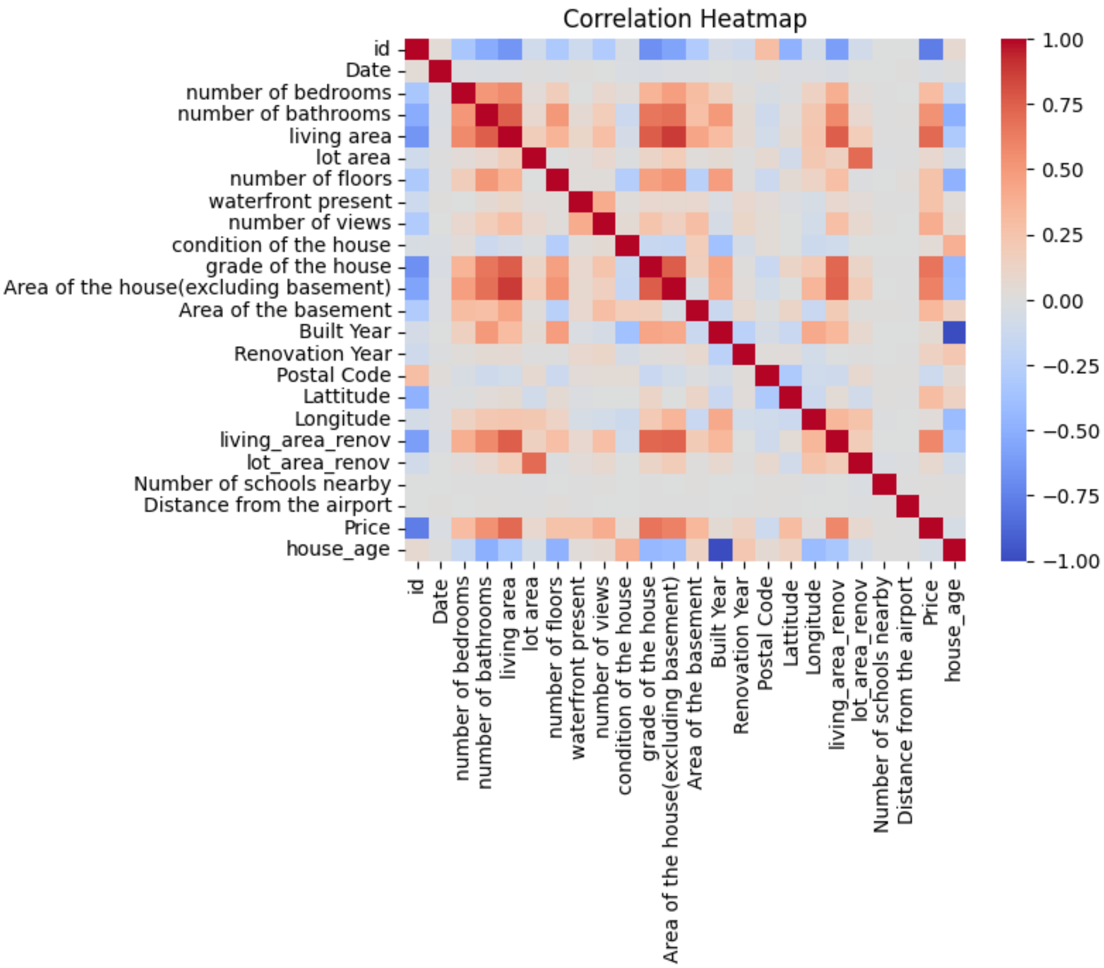
[ ]:

```
## BIVARIATE ANALYSIS ##
# Scatterplot for age and price
supermarket_sales_data["house_age"] = supermarket_sales_data["Built Year"].
  ↪max() - supermarket_sales_data["Built Year"]
sns.scatterplot(x="house_age", y="Price", data=supermarket_sales_data, alpha=0.
  ↪5)
plt.xlabel('House Age')
plt.ylabel('Price')
plt.title('Scatter Plot between House_Age and Price')
plt.show()
```



[ ]:

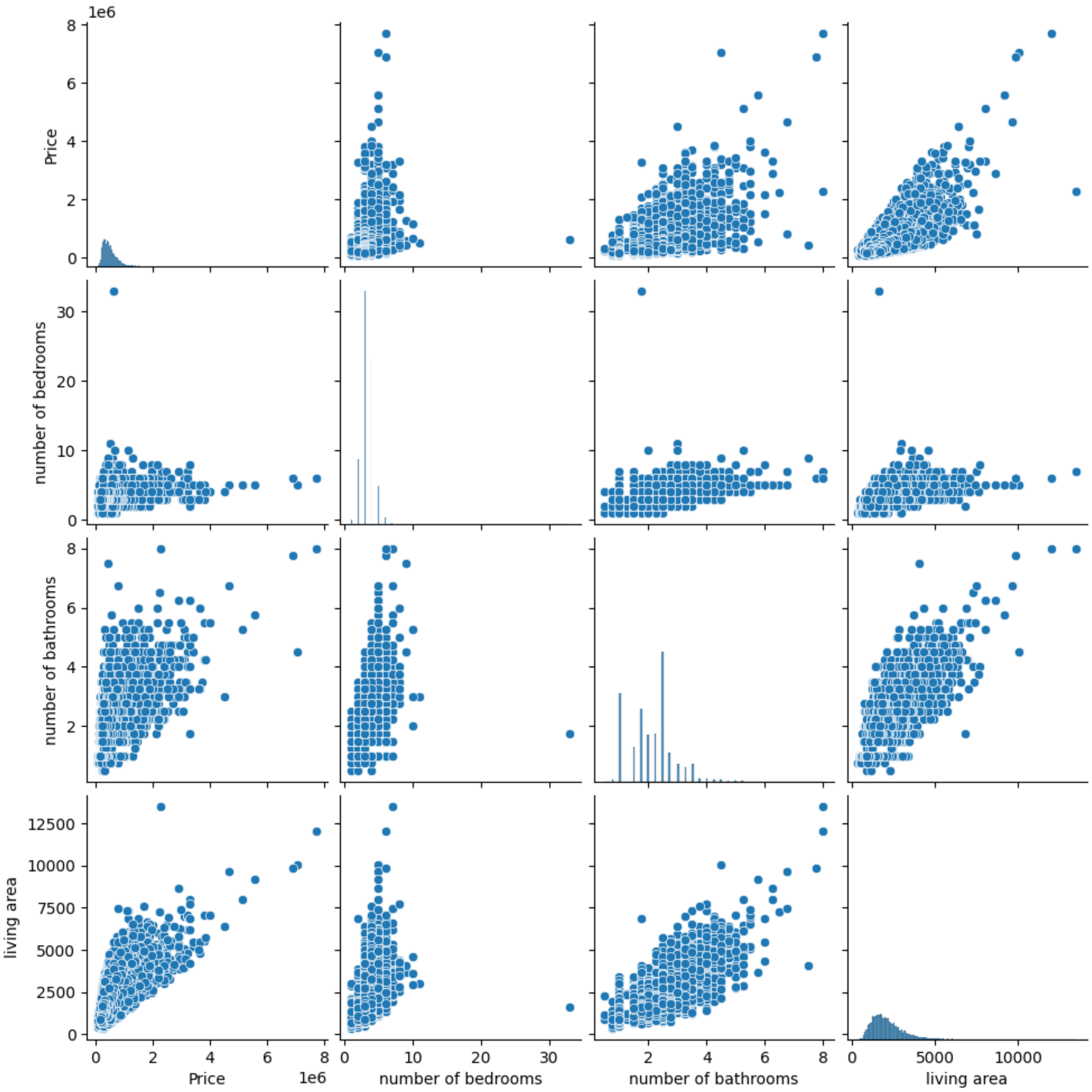
```
#Co-relation
correlation_matrix = supermarket_sales_data.corr()
sns.heatmap(correlation_matrix, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.figure(figsize=(50, 50))
plt.show()
```



```
<Figure size 5000x5000 with 0 Axes>
```

[ ]:

```
## MULTIVARIATE ANALYSIS ##
# Pair Plot
sns.pairplot(supermarket_sales_data[["Price", "number of bedrooms", "number of␣
↪bathrooms", "living area"]])
plt.show()
```



[ ]:

```
## DESCRIPTIVE STATISTICS ##
descriptive_stats = supermarket_sales_data.describe()
print(descriptive_stats)
```

```
                 id          Date  number of bedrooms  number of bathrooms  \
count  1.462000e+04  14620.000000        14620.000000         14620.000000
mean   6.762821e+09  42604.538646            3.379343             2.129583
std    6.237575e+03     67.347991            0.938719             0.769934
min    6.762810e+09  42491.000000            1.000000             0.500000
25%    6.762815e+09  42546.000000            3.000000             1.750000
50%    6.762821e+09  42600.000000            3.000000             2.250000
75%    6.762826e+09  42662.000000            4.000000             2.500000
max    6.762832e+09  42734.000000           33.000000             8.000000
```

```
        living area      lot area  number of floors  waterfront present  \
count  14620.000000  1.462000e+04      14620.000000        14620.000000
mean    2098.262996  1.509328e+04          1.502360            0.007661
std      928.275721  3.791962e+04          0.540239            0.087193
min      370.000000  5.200000e+02          1.000000            0.000000
25%     1440.000000  5.010750e+03          1.000000            0.000000
50%     1930.000000  7.620000e+03          1.500000            0.000000
75%     2570.000000  1.080000e+04          2.000000            0.000000
max    13540.000000  1.074218e+06          3.500000            1.000000

       number of views  condition of the house  …  Renovation Year  \
count     14620.000000            14620.000000  …     14620.000000
mean          0.233105                3.430506  …        90.924008
std           0.766259                0.664151  …       416.216661
min           0.000000                1.000000  …         0.000000
25%           0.000000                3.000000  …         0.000000
50%           0.000000                3.000000  …         0.000000
75%           0.000000                4.000000  …         0.000000
max           4.000000                5.000000  …      2015.000000

         Postal Code     Lattitude     Longitude  living_area_renov  \
count   14620.000000  14620.000000  14620.000000       14620.000000
mean   122033.062244     52.792848   -114.404007        1996.702257
std        19.082418      0.137522      0.141326         691.093366
min    122003.000000     52.385900   -114.709000         460.000000
25%    122017.000000     52.707600   -114.519000        1490.000000
50%    122032.000000     52.806400   -114.421000        1850.000000
75%    122048.000000     52.908900   -114.315000        2380.000000
max    122072.000000     53.007600   -113.505000        6110.000000

       lot_area_renov  Number of schools nearby  Distance from the airport  \
count    14620.000000              14620.000000               14620.000000
mean     12753.500068                  2.012244                  64.950958
std      26058.414467                  0.817284                   8.936008
min        651.000000                  1.000000                  50.000000
25%       5097.750000                  1.000000                  57.000000
50%       7620.000000                  2.000000                  65.000000
75%      10125.000000                  3.000000                  73.000000
max     560617.000000                  3.000000                  80.000000

              Price     house_age
count  1.462000e+04  14620.000000
mean   5.389322e+05     44.073598
std    3.675324e+05     29.493625
min    7.800000e+04      0.000000
25%    3.200000e+05     18.000000
50%    4.500000e+05     40.000000
```

```
75%    6.450000e+05     64.000000
max    7.700000e+06    115.000000

[8 rows x 24 columns]
```

```
[ ]: # Mode
     mode = supermarket_sales_data['Price'].mode()
     mode
```

```
[ ]: 0    450000
     Name: Price, dtype: int64
```

```
[ ]: ## HANDLING MISSING VALUES ##
     missing_values = supermarket_sales_data.isnull().sum()
     df_cleaned = supermarket_sales_data.dropna()
     missing_values
```

```
[ ]: id                                       0
     Date                                     0
     number of bedrooms                       0
     number of bathrooms                      0
     living area                              0
     lot area                                 0
     number of floors                         0
     waterfront present                       0
     number of views                          0
     condition of the house                   0
     grade of the house                       0
     Area of the house(excluding basement)    0
     Area of the basement                     0
     Built Year                               0
     Renovation Year                          0
     Postal Code                              0
     Lattitude                                0
     Longitude                                0
     living_area_renov                        0
     lot_area_renov                           0
     Number of schools nearby                 0
     Distance from the airport                0
     Price                                    0
     house_age                                0
     dtype: int64
```