Assignment-2

NAME : SASANK
REGNO : 21BCE9712

Tasks:-

: Dataset

1. Download the dataset

2. Load the dataset.

3. Perform the Below Visualizations. Univariate Analysis

Bi - Variate Analysis

- **Multivariate Analysis**

4. Perform descriptive statistics on the dataset.

5. Handle the Missing values

Loading Dataset

In [1]:

```
import io
import pandas as pd
#
```

```
               id   Date  number of bedrooms  number of bathrooms  \
0      6762810145  42491                   5                 2.50
1      6762810635  42491                   4                 2.50
2      6762810998  42491                   5                 2.75
3      6762812605  42491                   4                 2.50
4      6762812919  42491                   3                 2.00
...           ...    ...                 ...                  ...
14615  6762830250  42734                   2                 1.50
14616  6762830339  42734                   3                 2.00
14617  6762830618  42734                   2                 1.00
14618  6762830709  42734                   4                 1.00
14619  6762831463  42734                   3                 1.00

       living area  lot area  number of floors  waterfront present  \
0             3650      9050               2.0                   0
1             2920      4000               1.5                   0
2             2910      9480               1.5                   0
3             3310     42998               2.0                   0
4             2710      4500               1.5                   0
...            ...       ...               ...                 ...
14615         1556     20000               1.0                   0
14616         1680      7000               1.5                   0
14617         1070      6120               1.0                   0
14618         1030      6621               1.0                   0
14619          900      4770               1.0                   0

       number of views  condition of the house  ...  Built Year  \
0                    4                       5  ...        1921
1                    0                       5  ...        1909
2                    0                       3  ...        1939
3                    0                       3  ...        2001
4                    0                       4  ...        1929
...                ...                     ...  ...         ...
14615                0                       4  ...        1957
14616                0                       4  ...        1968
14617                0                       3  ...        1962
14618                0                       4  ...        1955
14619                0                       3  ...        1969

       Renovation Year  Postal Code  Lattitude  Longitude  living_area
_renov  \
0                    0       122003    52.8645   -114.557
2880
1                    0       122004    52.8878   -114.470
2470
2                    0       122004    52.8852   -114.468
2940
3                    0       122005    52.9532   -114.321
3350
4                    0       122006    52.9047   -114.485
2060
...                ...          ...        ...        ...
...
14615                0       122066    52.6191   -114.472
2250
14616                0       122072    52.5075   -114.393
1540
14617                0       122056    52.7289   -114.507
1130
14618                0       122042    52.7157   -114.411
1420
```

.

```
14619             2009       122018    52.5338   -114.552
900

       lot_area_renov  Number of schools nearby  Distance from the air
port \
0                5400                         2
58
1                4000                         2
51
2                6600                         1
53
3               42847                         3
76
4                4500                         1
51
...               ...                       ...
...
14615           17286                         3
76
14616            7480                         3
59
14617            6120                         2
64
14618            6631                         3
54
14619            3480                         2
55

         Price
0      2380000
1      1400000
2      1200000
3       838000
4       805000
...        ...
14615   221700
14616   219200
14617   209000
14618   205000
14619   146000

[14620 rows x 23 columns]
```

.

In [4]:

```
df.head()
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14620 entries, 0 to 14619
Data columns (total 23 columns):
 #   Column                                 Non-Null Count  Dtype
---  ------                                 --------------  -----
 0   id                                     14620 non-null  int64
 1   Date                                   14620 non-null  int64
 2   number of bedrooms                     14620 non-null  int64
 3   number of bathrooms                    14620 non-null  float64
 4   living area                            14620 non-null  int64
 5   lot area                               14620 non-null  int64
 6   number of floors                       14620 non-null  float64
 7   waterfront present                     14620 non-null  int64
 8   number of views                        14620 non-null  int64
 9   condition of the house                 14620 non-null  int64
 10  grade of the house                     14620 non-null  int64
 11  Area of the house(excluding basement)  14620 non-null  int64
 12  Area of the basement                   14620 non-null  int64
 13  Built Year                             14620 non-null  int64
 14  Renovation Year                        14620 non-null  int64
 15  Postal Code                            14620 non-null  int64
 16  Lattitude                              14620 non-null  float64
 17  Longitude                              14620 non-null  float64
 18  living_area_renov                      14620 non-null  int64
 19  lot_area_renov                         14620 non-null  int64
 20  Number of schools nearby               14620 non-null  int64
 21  Distance from the airport              14620 non-null  int64
 22  Price                                  14620 non-null  int64
dtypes: float64(4), int64(19)
memory usage: 2.6 MB
```

Out[2]:

	id	Date	number of bedrooms	number of bath	living area	lot area
count	1.462000e+04	14620.000000	14620.000000	14620.		
mean	6.762821e+09	42604.538646	3.379343	2.		
std	6.237575e+03	67.347991	0.938719	0.		
min	6.762810e+09	42491.000000	1.000000	0.		

8 rows × 23 columns

In [5]:

```
df.shape
df.dtypes
```

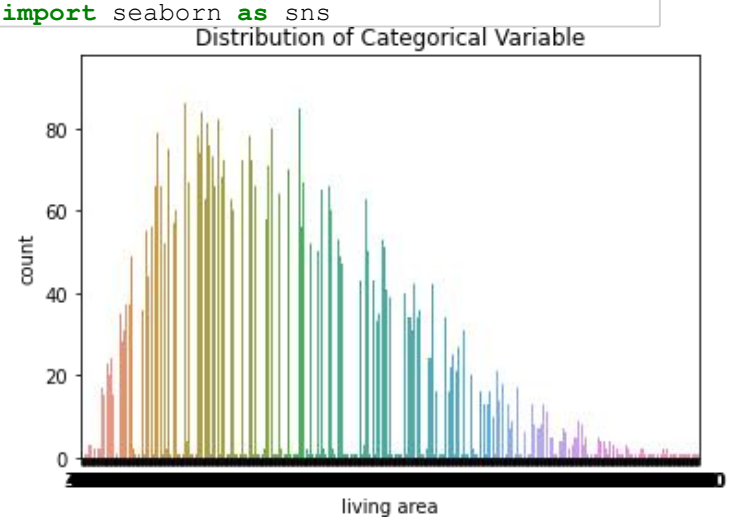
Out[3]:

```
id                                          int64
Date                                        int64
number of bedrooms                          int64
number of bathrooms                       float64
living area                                 int64
lot area                                    int64
number of floors                          float64
waterfront present                          int64
number of views                             int64
condition of the house                      int64
grade of the house                          int64
Area of the house(excluding basement)       int64
Area of the basement                        int64
Built Year                                  int64
Renovation Year                             int64
Postal Code                                 int64
Lattitude                                 float64
Longitude                                 float64
living_area_renov                           int64
lot_area_renov                              int64
Number of schools nearby                    int64
Distance from the airport                   int64
Price                                       int64
dtype: object
```

Univariate Analysis

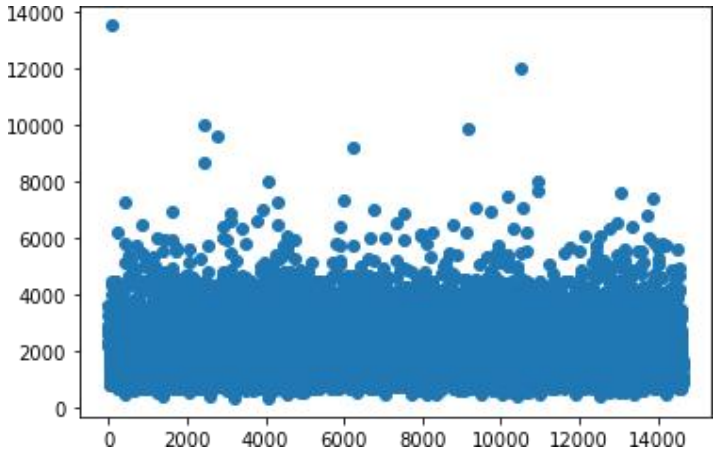
In [4]:

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



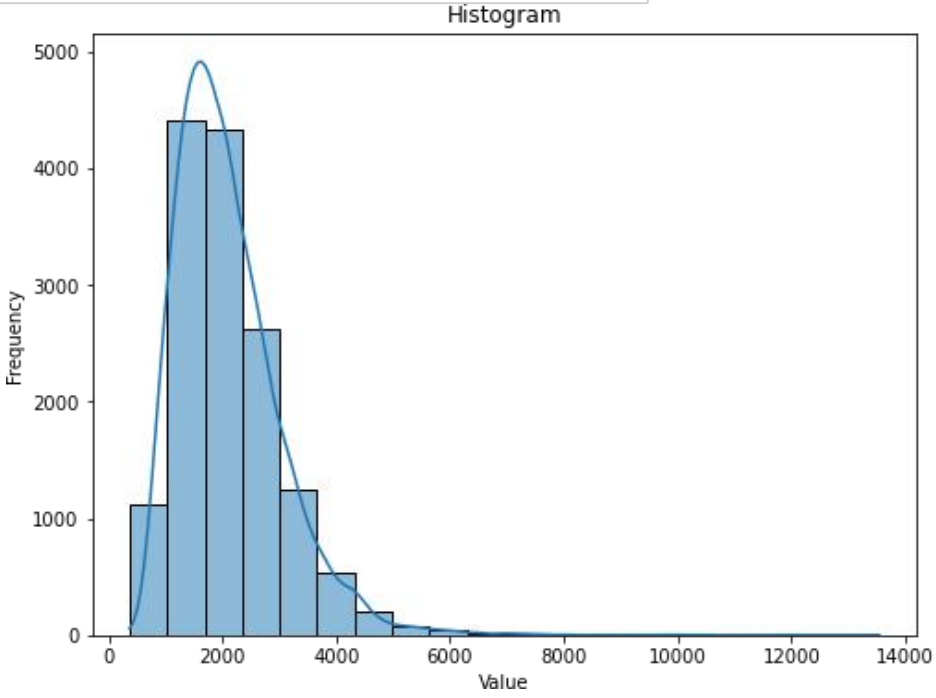
In [5]:

```
plt.scatter(x=df.index,y=df['living area'])
plt.show()
```



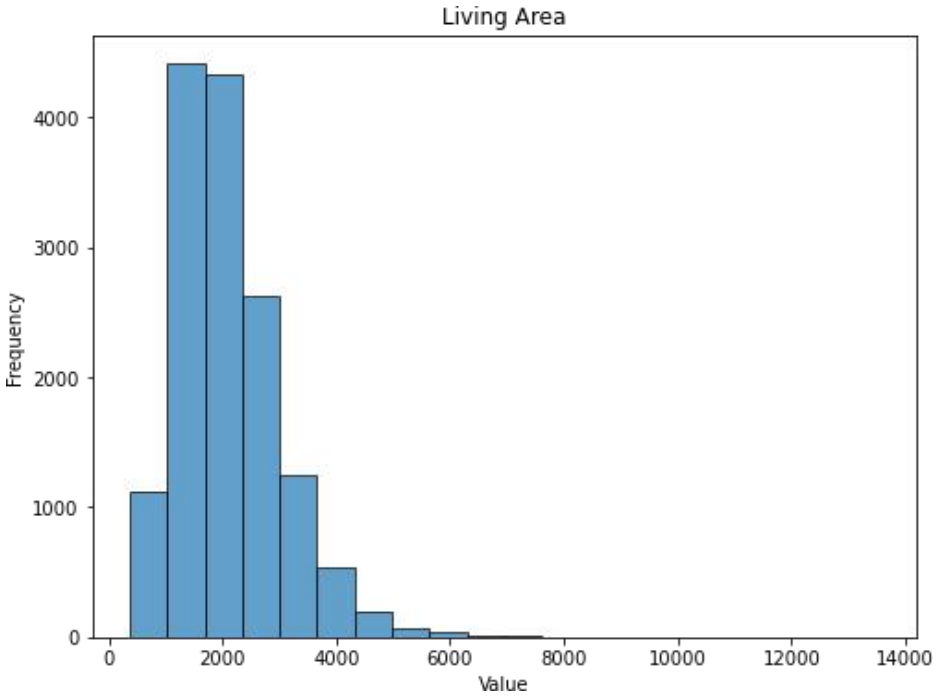
In [6]:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(8,6))
sns.hi
```



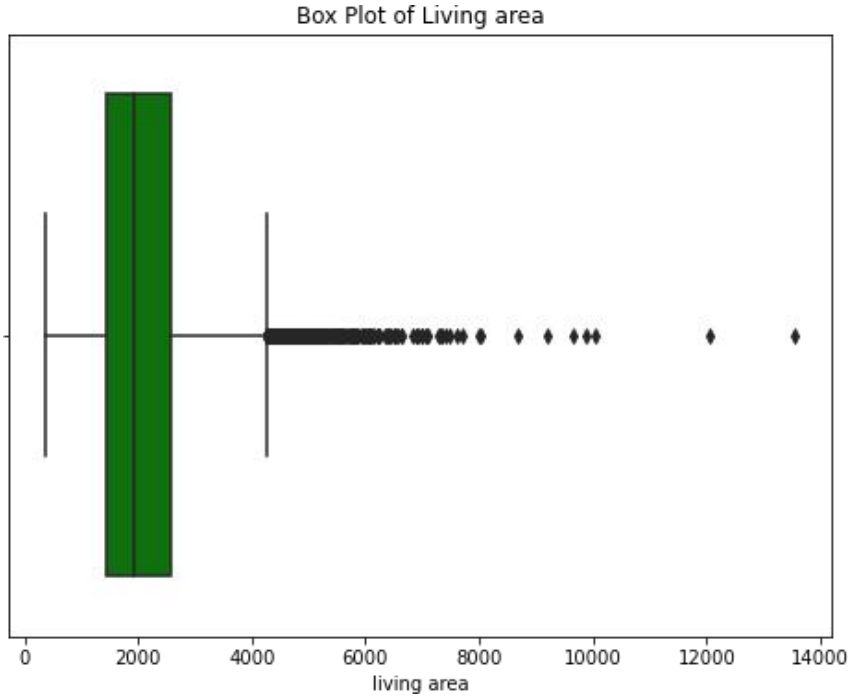
In [7]:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(8,6))
plt.hist(df['living area'],bins=20,edgecolor='k',alpha=0.7)
plt.title('Living Area')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```



In [8]:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(8,6))
sns.boxplot(x=df['living area'],color='green')
plt.title('Box Plot of Living area')
plt.show()
```

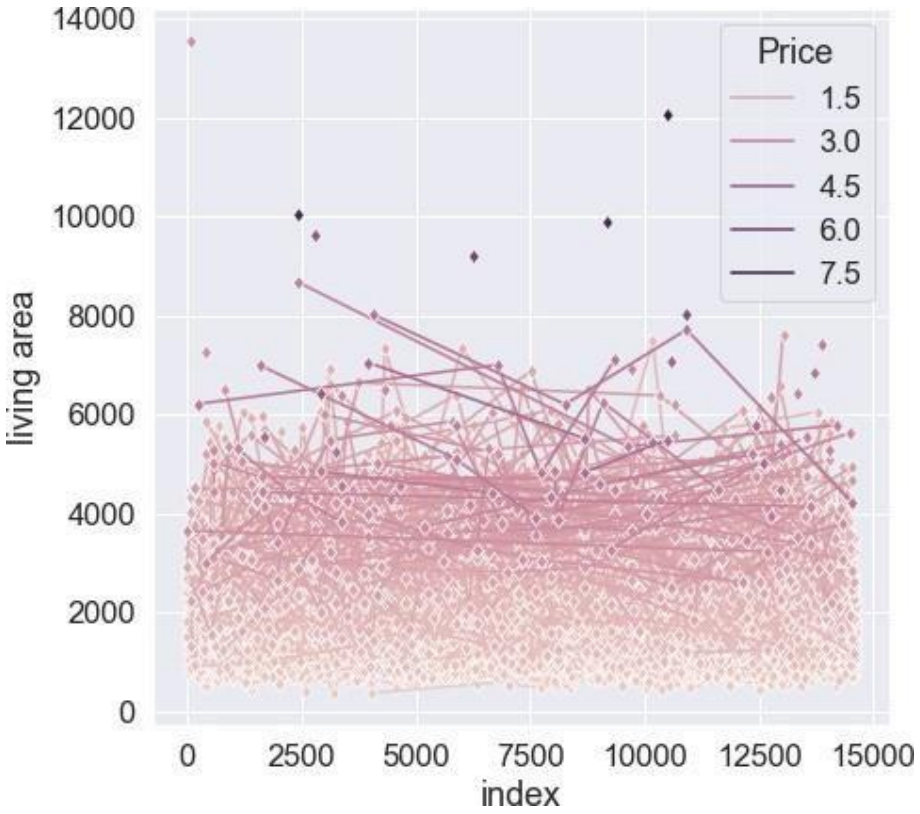


In [10

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(rc={'figure.figsize':(7,7)})
sns.set(font_scale=1.5)
fig=sns.lineplot(x=df.index,y=df['living area'],marker='d',data=df,hue=df['Price'])
fig.set(xlabel='index')
```

Out[9]:

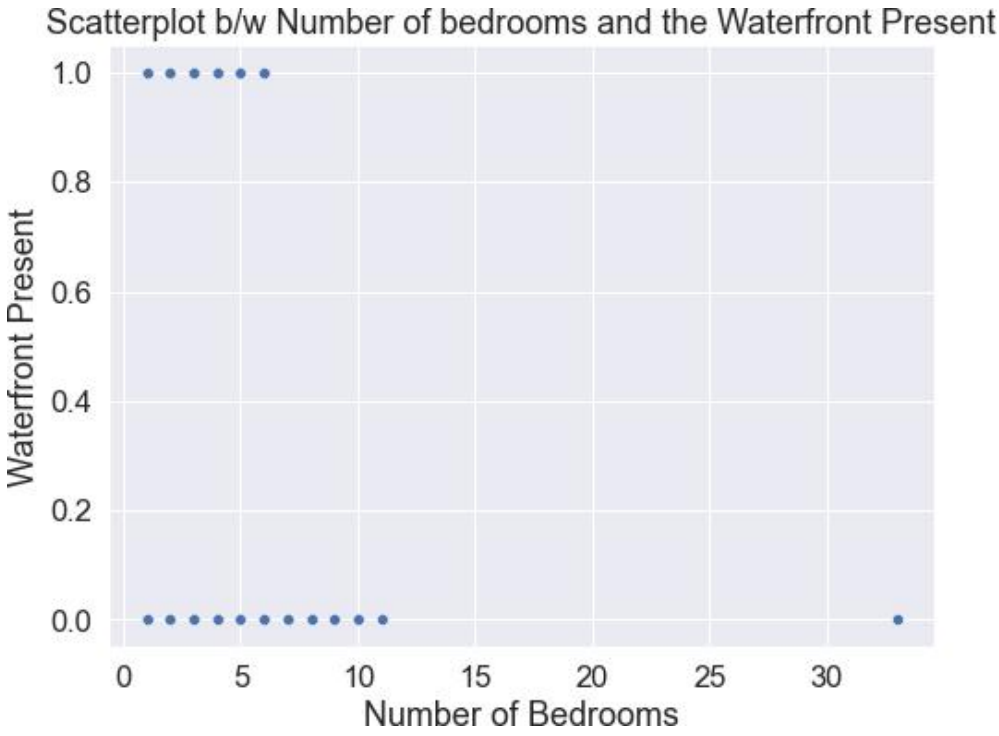
```
[Text(0.5, 0, 'index')]
```



Bi variate Analysis

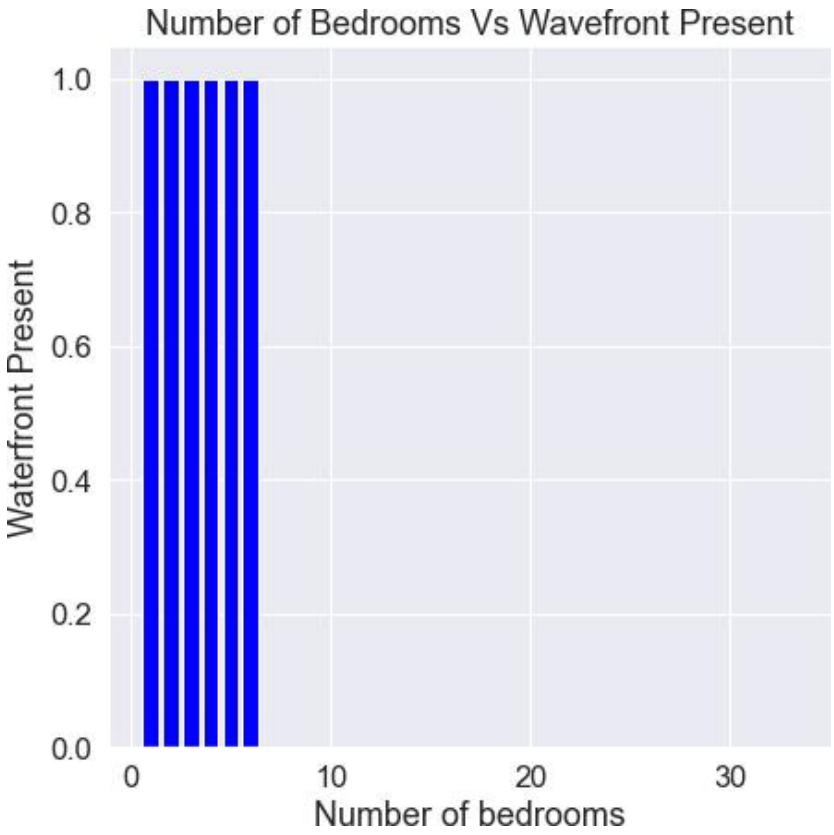
In [10]:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(8,6))
sns.scatterplot(x='number of bedrooms',y='waterfront present',data=df)
plt.title('Scatterplot b/w Number of bedrooms and the Waterfront Present')
plt.xlabel('Number of Bedrooms')
plt.ylabel('Waterfront Present')
plt.show()
```



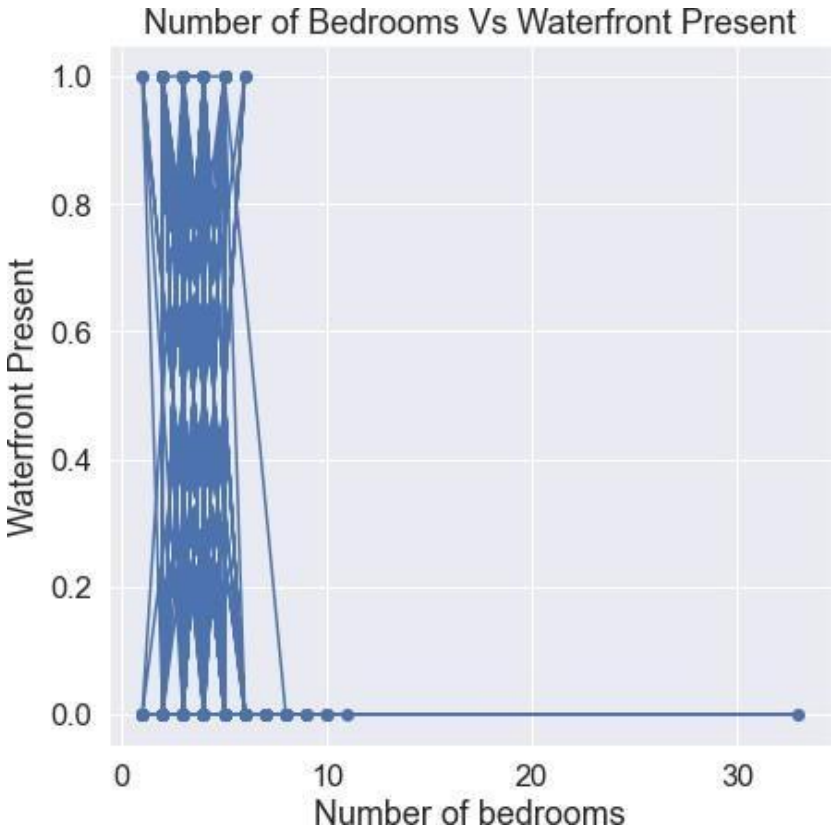
In [11]:

```
plt.bar(df['number of bedrooms'],df['waterfront present'],color="blue")
plt.xlabel('Number of bedrooms')
plt.ylabel('Waterfront Present')
plt.title("Number of Bedrooms Vs Wavefront Present")
plt.show()
```



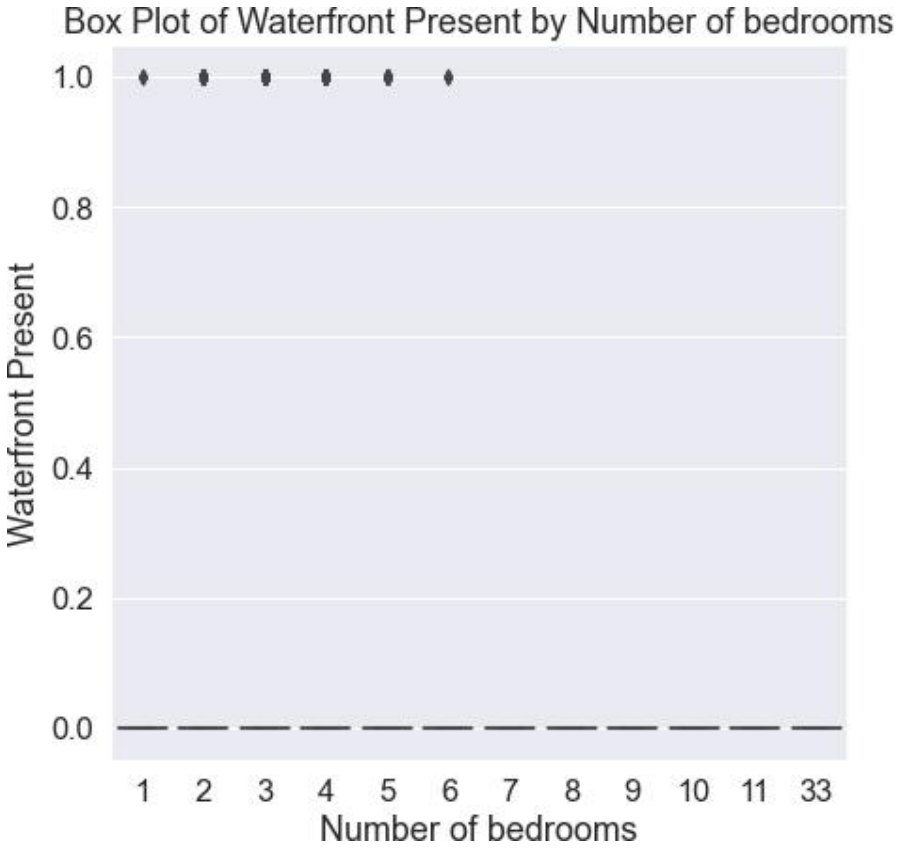
In [12

```
plt.plot(df['number of bedrooms'],df['waterfront present'],marker='o',linestyle='-')
plt.xlabel('Number of bedrooms')
plt.ylabel('Waterfront Present')
plt.title("Number of Bedrooms Vs Waterfront Present")
plt.show()
```



In [13

```
sns.boxplot(x='number of bedrooms', y='waterfront present', data=df)
plt.xlabel('Number of bedrooms')
plt.ylabel('Waterfront Present')
plt.title('Box Plot of Waterfront Present by Number of bedrooms')
plt.show()
```



In [14]:

```
correlation_coefficient = df['number of
```

```
Correlation coefficient: -0.006256664357638965
```

In [15]:

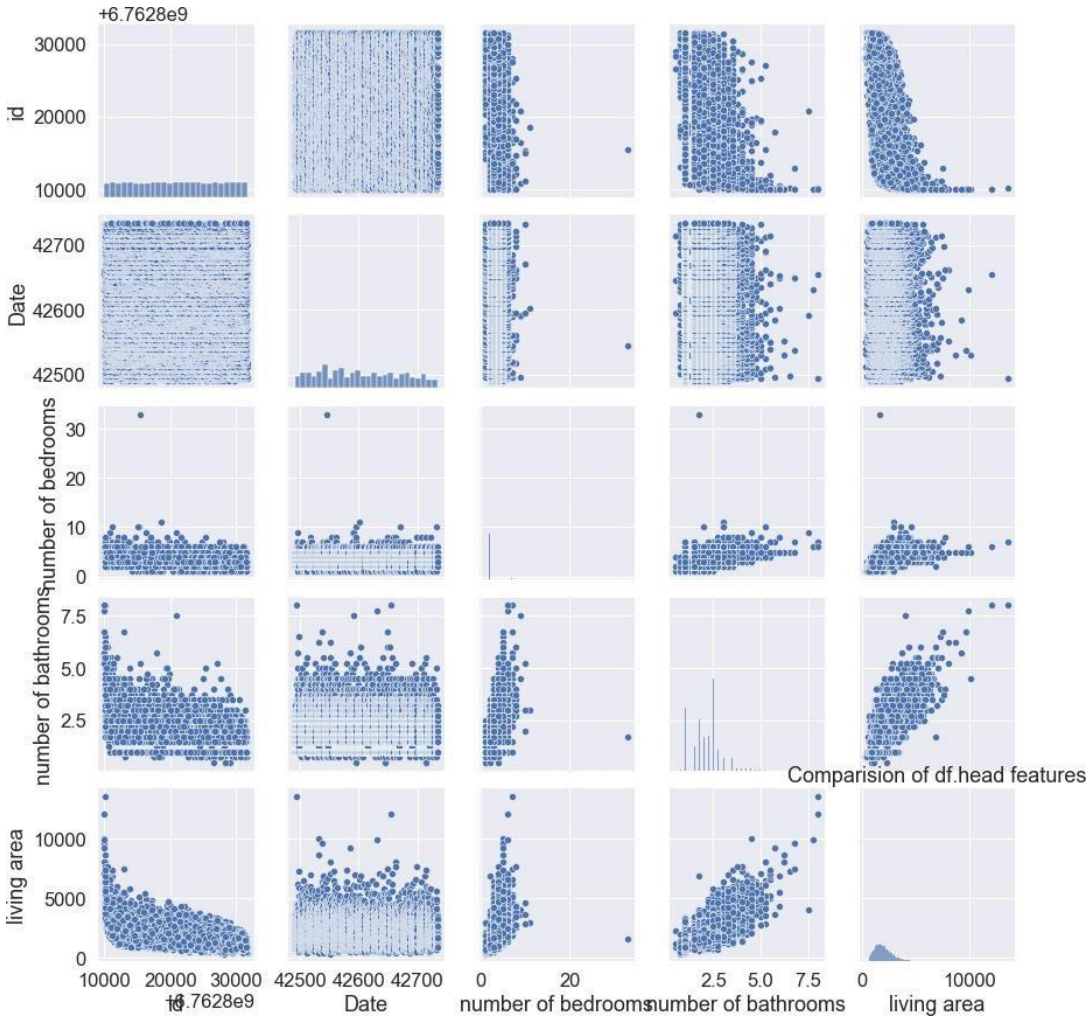
```
from scipy.stats import chi2_contingency
contingency_table =
```

```
Chi-squared statistic: 9.620884386637231
p-value: 0.564776259787771
```

Multivariate Analysis

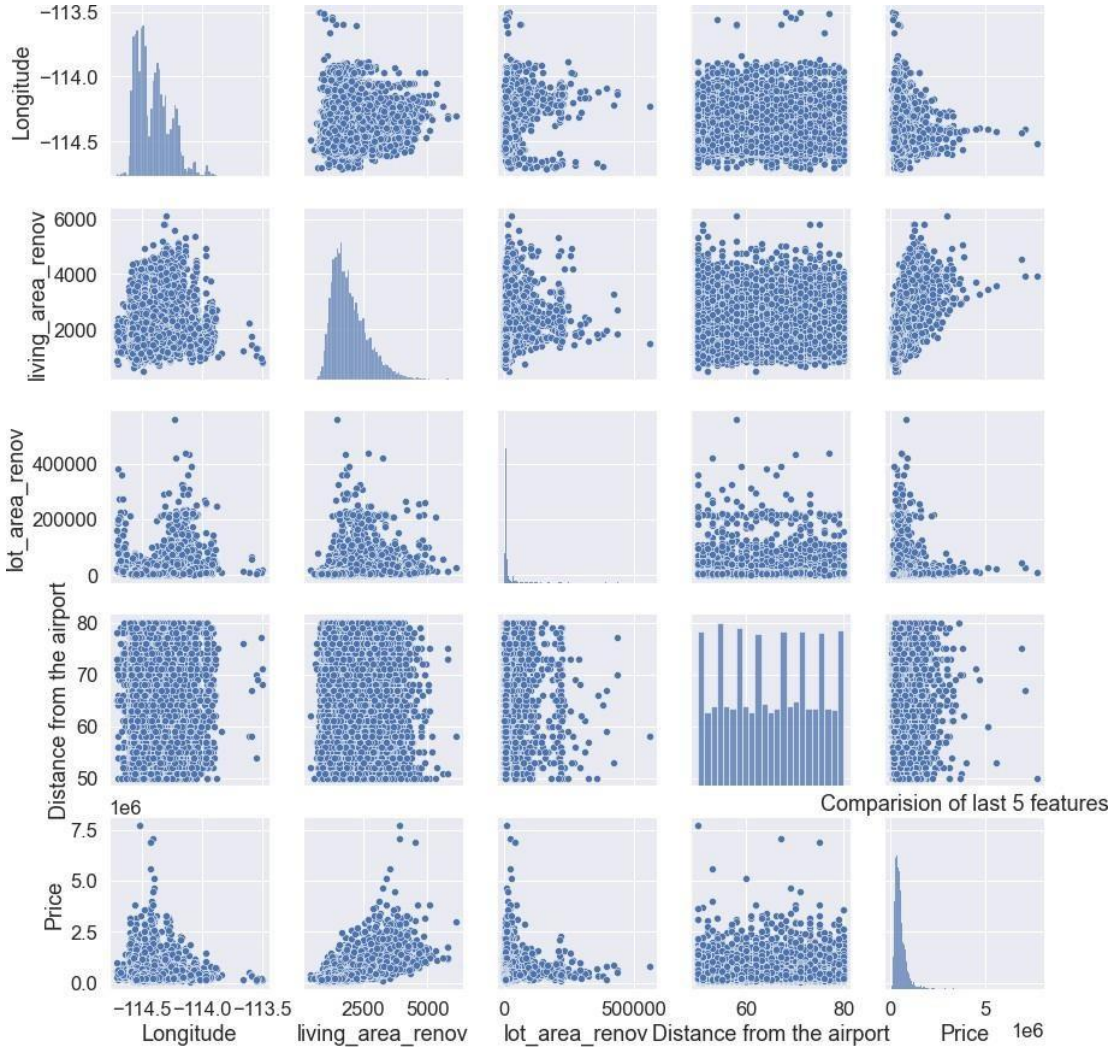
In [14

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.pairplot(df,vars=['id','Date','number of bedrooms','number of bathrooms','living
plt.title('Comparision of df.head features')
plt.show()
```



In [15

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.pairplot(df,vars=['Longitude','living_area_renov','lot_area_renov','Distance fro
plt.title('Comparision of last 5 features')
plt.show()
```



Descriptive Statistics

In [25]:

```
print(df.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 Co
de  \
0                       5  ...        1921                0       1220
03
1                       5  ...        1909                0       1220
04
2                       3  ...        1939                0       1220
04
3                       3  ...        2001                0       1220
05
4                       4  ...        1929                0       1220
06

   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]
```

In [21]:

```
unique_values = df['Price'].nunique()
print(f'Number of unique values: {unique_values}')
mean_value = df['Price'].mean()
print(f'Mean: {mean_value}')
mode_value = df['Price'].mode().values[0]
print(f'Mode: {mode_value}')
median_value = df['Price'].median()
print(f'Median: {median_value}')
percentile_25 = df['Price'].quantile(0.25)
print(f'Percentile_25: {percentile_25}')
percentile_50 = df['Price'].quantile(0.50)
print(f'Percentile_50: {percentile_50}')
percentile_75 = df['Price'].quantile(0.75)
print(f'Percentile_75: {percentile_75}')
variance = df['Price'].var()
print(f'Variance: {variance}')
std_deviation = df['Price'].std()
print(f'Standard deviation: {std_deviation}')
skewness = df['Price'].skew()
print(f'skewness: {skewness}')
kurtosis = df['Price'].kurtosis()
print(f'kurtosis: {kurtosis}')
```

```
Number of unique values: 2901Mean:
538932.2183310534
Mode: 450000
Median: 450000.0
Percentile_25: 320000.0
Percentile_50: 450000.0
Percentile_75: 645000.0
Variance: 135080050939.43213
Standard deviation: 367532.3808039669
skewness: 4.269297720707116
kurtosis: 40.32191815363438
```

In [26]:

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14620 entries, 0 to 14619
Data columns (total 23 columns):
 #   Column                                 Non-Null Count  Dtype  
---  ---------                              --------------  -----  
 0   id                                     14620 non-null  int64  
 1   Date                                   14620 non-null  int64  
 2   number of bedrooms                     14620 non-null  int64  
 3   number of bathrooms                    14620 non-null  float64
 4   living area                            14620 non-null  int64  
 5   lot area                               14620 non-null  int64  
 6   number of floors                       14620 non-null  float64
 7   waterfront present                     14620 non-null  int64  
 8   number of views                        14620 non-null  int64  
 9   condition of the house                 14620 non-null  int64  
 10  grade of the house                     14620 non-null  int64  
 11  Area of the house(excluding basement)  14620 non-null  int64  
 12  Area of the basement                   14620 non-null  int64  
 13  Built Year                             14620 non-null  int64  
 14  Renovation Year                        14620 non-null  int64  
 15  Postal Code                            14620 non-null  int64  
 16  Lattitude                              14620 non-null  float64
 17  Longitude                              14620 non-null  float64
 18  living_area_renov                      14620 non-null  int64  
 19  lot_area_renov                         14620 non-null  int64  
 20  Number of schools nearby               14620 non-null  int64  
 21  Distance from the airport              14620 non-null  int64  
 22  Price                                  14620 non-null  int64  
dtypes: float64(4), int64(19)
memory usage: 2.6 MB
None
```

In [22]:

```
unique_values = df['living area'].nunique()
print(f'Number of unique values: {unique_values}')
mean_value = df['living area'].mean()
print(f'Mean: {mean_value}')
mode_value = df['living area'].mode().values[0]
print(f'Mode: {mode_value}')
median_value = df['living area'].median()
print(f'Median: {median_value}')
percentile_25 = df['living area'].quantile(0.25)
print(f'Percentile_25: {percentile_25}')
percentile_50 = df['living area'].quantile(0.50)
print(f'Percentile_50: {percentile_50}')
percentile_75 = df['living area'].quantile(0.75)
print(f'Percentile_75: {percentile_75}')
variance = df['living area'].var()
print(f'Variance: {variance}')
std_deviation = df['living area'].std()
print(f'Standard deviation: {std_deviation}')
skewness = df['living area'].skew()
print(f'skewness: {skewness}')
kurtosis = df['living area'].kurtosis()
print(f'kurtosis: {kurtosis}')
```

```
Number of unique values: 865Mean:
2098.262995896033
Mode: 1400
Median: 1930.0
Percentile_25: 1440.0
Percentile_50: 1930.0
Percentile_75: 2570.0
Variance: 861695.8146098064
Standard deviation: 928.2757212217749
skewness: 1.538336624376669
kurtosis: 6.0736171462473205
```

In [27]:

```
                id          Date  number of bedrooms  number of bathr
ooms  \
count  1.462000e+04  14620.000000        14620.000000         14620.00
0000
mean   6.762821e+09  42604.538646            3.379343             2.12
9583
std    6.237575e+03     67.347991            0.938719             0.76
9934
min    6.762810e+09  42491.000000            1.000000             0.50
0000
25%    6.762815e+09  42546.000000            3.000000             1.75
0000
50%    6.762821e+09  42600.000000            3.000000             2.25
0000
75%    6.762826e+09  42662.000000            4.000000             2.50
0000
max    6.762832e+09  42734.000000           33.000000             8.00
0000

        living area      lot area  number of floors  waterfront presen
t  \
count  14620.000000  1.462000e+04      14620.000000        14620.00000
0
mean    2098.262996  1.509328e+04          1.502360            0.00766
1
std      928.275721  3.791962e+04          0.540239            0.08719
3
min      370.000000  5.200000e+02          1.000000            0.00000
0
25%     1440.000000  5.010750e+03          1.000000            0.00000
0
50%     1930.000000  7.620000e+03          1.500000            0.00000
0
75%     2570.000000  1.080000e+04          2.000000            0.00000
0
max    13540.000000  1.074218e+06          3.500000            1.00000
0

       number of views condition of the house ...  Built Year  \
count     14620.000000            14620.000000.....14620.000000
mean          0.233105                3.430506.....1970.926402
std           0.766259                0.664151  ...     29.493625
min           0.000000                1.000000.....1900.000000
25%           0.000000                3.000000  ...   1951.000000
50%           0.000000                3.000000  ...   1975.000000
75%           0.000000                4.000000  ...   1997.000000
max           4.000000                5.000000.....2015.000000

       Renovation Year    Postal Code     Lattitude     Longitude \
count     14620.000000   14620.000000  14620.000000  14620.000000
mean         90.924008  122033.062244     52.792848   -114.404007
std         416.216661      19.082418      0.137522      0.141326
min           0.000000  122003.000000     52.385900   -114.709000
25%           0.000000  122017.000000     52.707600   -114.519000
50%           0.000000  122032.000000     52.806400   -114.421000
75%           0.000000  122048.000000     52.908900   -114.315000
max        2015.000000  122072.000000     53.007600   -113.505000

       living_area_renov lot_area_renov Number of schools nearby \
count       14620.000000    14620.000000              14620.000000
mean         1996.702257    12753.500068                  2.012244
```

```
std            691.093366    26058.414467                    0.817284
min            460.000000      651.000000                    1.000000
25%           1490.000000     5097.750000                    1.000000
50%           1850.000000     7620.000000                    2.000000
75%           2380.000000    10125.000000                    3.000000
max           6110.000000   560617.000000                    3.000000

       Distance from the airport         Price
count                14620.000000  1.462000e+04
mean                    64.950958  5.389322e+05
std                      8.936008  3.675324e+05
min                     50.000000  7.800000e+04
25%                     57.000000  3.200000e+05
50%                     65.000000  4.500000e+05
75%                     73.000000  6.450000e+05
max                     80.000000  7.700000e+06

[8 rows x 23 columns]
```

In [23]:

```
unique_values = df['lot area'].nunique()
print(f'Number of unique values: {unique_values}')
mean_value = df['lot area'].mean()
print(f'Mean: {mean_value}')
mode_value = df['lot area'].mode().values[0]
print(f'Mode: {mode_value}')
median_value = df['lot area'].median()
print(f'Median: {median_value}')
percentile_25 = df['lot area'].quantile(0.25)
print(f'Percentile_25: {percentile_25}')
percentile_50 = df['lot area'].quantile(0.50)
print(f'Percentile_50: {percentile_50}')
percentile_75 = df['lot area'].quantile(0.75)
print(f'Percentile_75: {percentile_75}')
variance = df['lot area'].var()
print(f'Variance: {variance}')
std_deviation = df['lot area'].std()
print(f'Standard deviation: {std_deviation}')
skewness = df['lot area'].skew()
print(f'skewness: {skewness}')
kurtosis = df['lot area'].kurtosis()
print(f'kurtosis: {kurtosis}')
```

```
Number of unique values: 7451Mean:
15093.281121751026
Mode: 5000
Median: 7620.0
Percentile_25: 5010.75
Percentile_50: 7620.0
Percentile_75: 10800.0
Variance: 1437897679.806871
Standard deviation: 37919.62130357938
skewness: 10.155206088640242
kurtosis: 164.75727344890643
```

Checking and Handling the Null values if present

In [24]:

```
df.isnull()
```

Out[24]:

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of	waterfront present	number of	condition of the
0	False	False	False	False	False	False				
1	False	False	False	False	False	False				
2	False	False	False	False	False	False				
3	False	False	False	False	False	False				
4	False	False	False	False	False	False				
...	...	...	...	...	...	...				

14620 rows × 23 columns

In [ ]:

In [ ]:

In [ ]:

In [ ]: