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

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_
```

```
df=pd.read_csv('/content/drive/MyDrive/insurance.csv')
df.head()
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0

5 rows × 23 columns

```
df.tail()
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
14615	6762830250	42734	2	1.5	1556	20000	1.0	0	0	4	...	1957	
14616	6762830339	42734	3	2.0	1680	7000	1.5	0	0	4	...	1968	
14617	6762830618	42734	2	1.0	1070	6120	1.0	0	0	3	...	1962	
14618	6762830709	42734	4	1.0	1030	6621	1.0	0	0	4	...	1955	
14619	6762831463	42734	3	1.0	900	4770	1.0	0	0	3	...	1969	

5 rows × 23 columns

```
df.shape
```

```
(14620, 23)
```

```
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
```

```
df.isnull().any()
```

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

Visualization

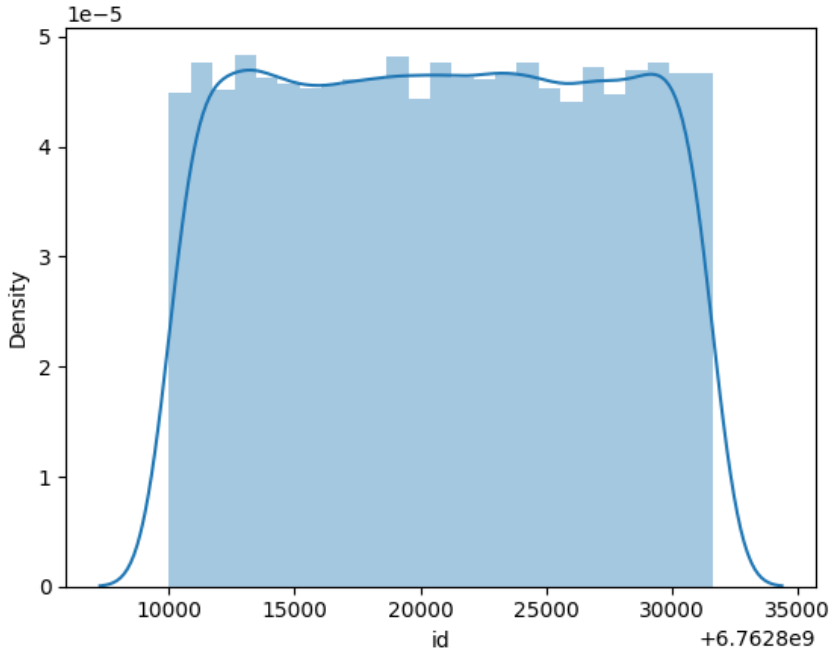
Univarient Analysis

```
sns.distplot(df.Price)
```

```
<ipython-input-17-5e080168c38c>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df.Price)
<Axes: xlabel='Price', ylabel='Density'>
```

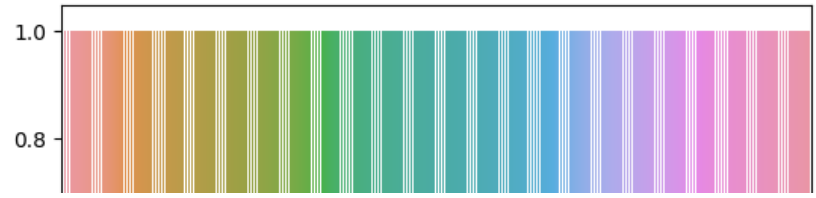

```
sns.distplot(df.id)
```

```
<ipython-input-8-ae12bbc1c629>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df.id)
<Axes: xlabel='id', ylabel='Density'>
```



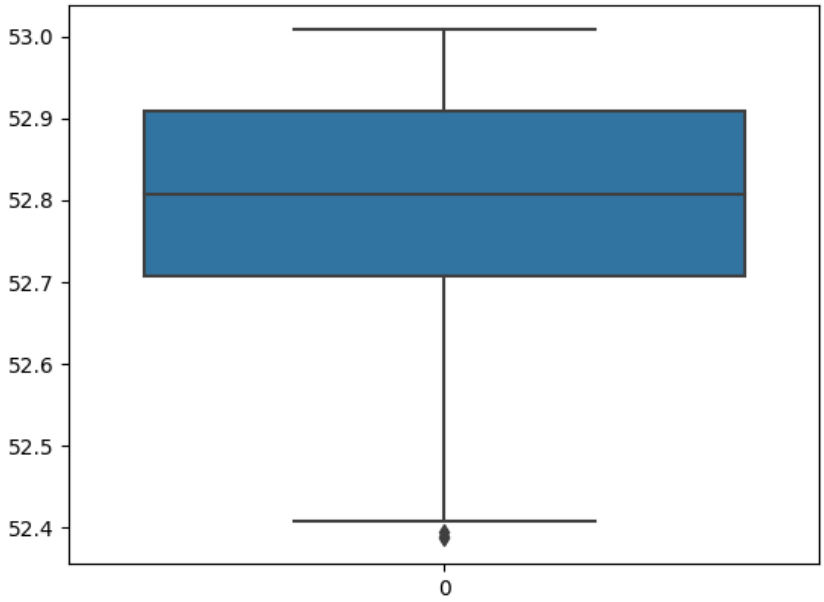

```
sns.barplot(x=df.id.value_counts().index,y=df.id.value_counts())
```

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




```
sns.boxplot(df.Lattitude)
```

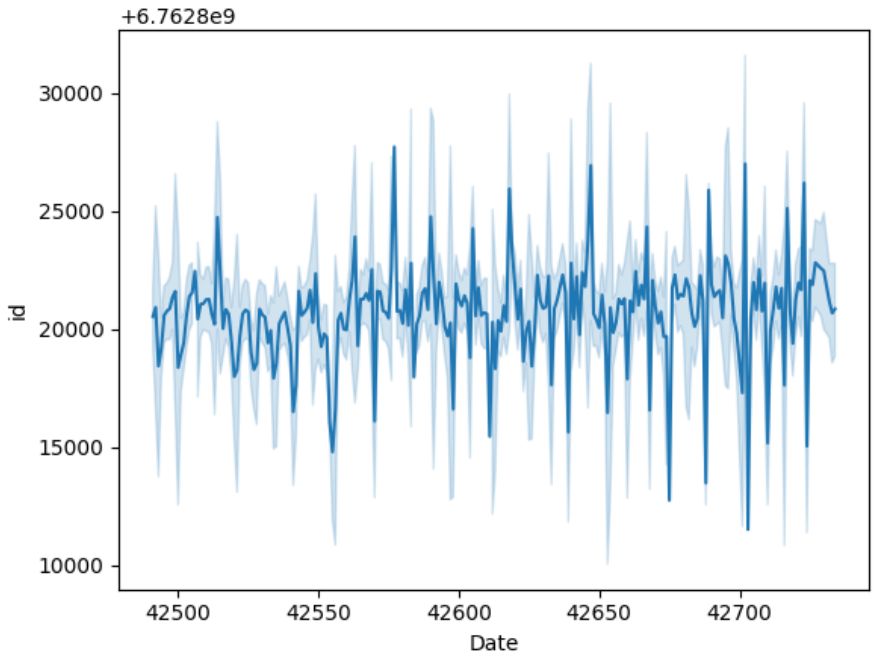
```
<Axes: >
```



Bivarient

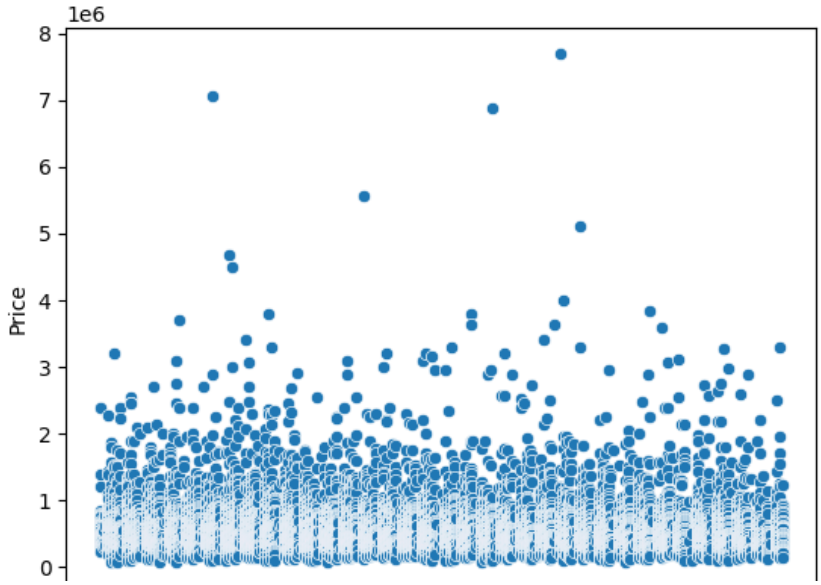
```
sns.lineplot(x=df.Date,y=df.id)
```

```
<Axes: xlabel='Date', ylabel='id'>
```




```
sns.scatterplot(x=df.Date,y=df.Price)
```

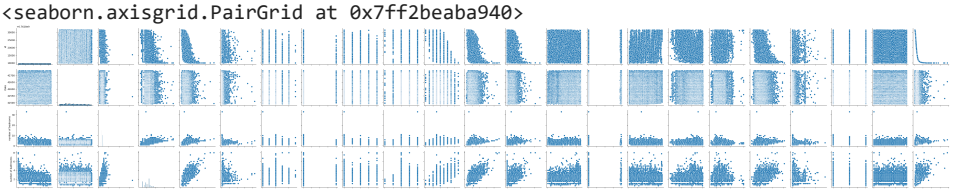
```
<Axes: xlabel='Date', ylabel='Price'>
```



Multivarient Analysis

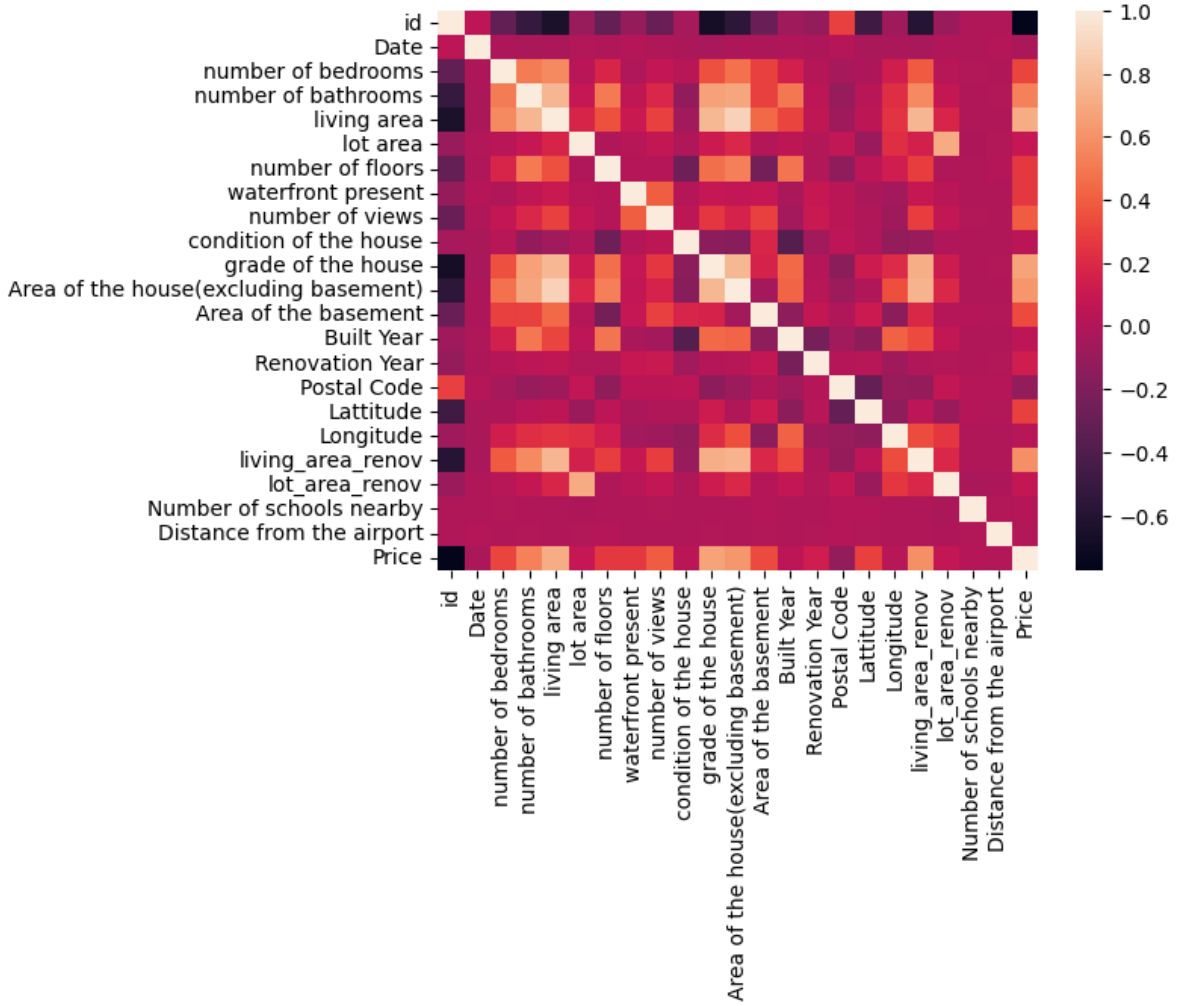
```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x7ff2beaba940>
```



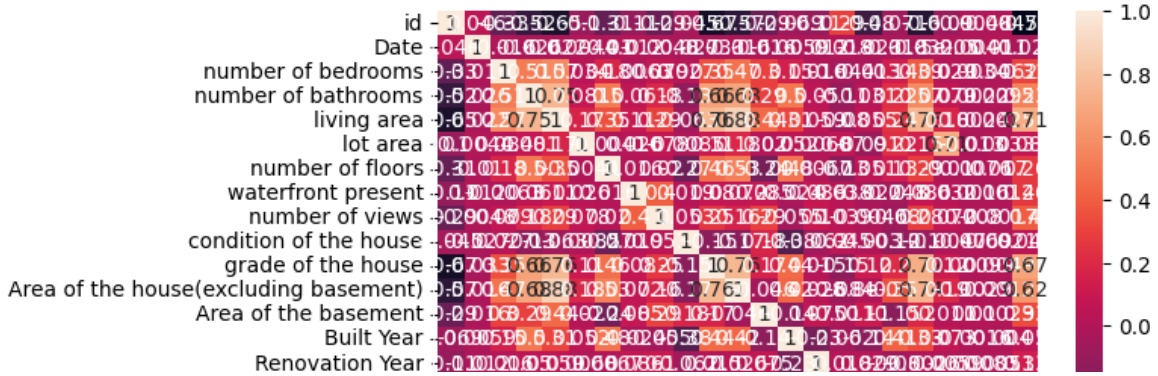
```
sns.heatmap(df.corr())
```

```
<Axes: >
```




```
sns.heatmap(df.corr(),annot=True)
```

```
<Axes: >
```



Descriptive statistic on the dataset


```
df.describe()   #descriptive statistics
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present
count	1.462000e+04	14620.000000	14620.000000	14620.000000	14620.000000	1.462000e+04	14620.000000	14620.000000
mean	6.762821e+09	42604.538646	3.379343	2.129583	2098.262996	1.509328e+04	1.502360	0.007661
std	6.237575e+03	67.347991	0.938719	0.769934	928.275721	3.791962e+04	0.540239	0.087193
min	6.762810e+09	42491.000000	1.000000	0.500000	370.000000	5.200000e+02	1.000000	0.000000
25%	6.762815e+09	42546.000000	3.000000	1.750000	1440.000000	5.010750e+03	1.000000	0.000000
50%	6.762821e+09	42600.000000	3.000000	2.250000	1930.000000	7.620000e+03	1.500000	0.000000
75%	6.762826e+09	42662.000000	4.000000	2.500000	2570.000000	1.080000e+04	2.000000	0.000000
max	6.762832e+09	42734.000000	33.000000	8.000000	13540.000000	1.074218e+06	3.500000	1.000000

8 rows × 23 columns

```
df.head()
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Renova
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	

5 rows × 23 columns

```
df.living_area_renov.unique()
```

```
array([2880, 2470, 2940, 3350, 2060, 2380, 3320, 1570, 2010, 2320, 2820,
       1910, 2390, 2410, 1300, 2730, 1860, 4050, 2570, 2200, 2590, 2860,
       1090, 3000, 1340, 2780, 2080, 2260, 2990, 1560, 1320, 1850, 1150,
       1770, 2340, 1680, 1260, 1450, 2070, 2290, 1960, 2830, 1440, 1790,
       1160, 1480, 1100, 2280, 1590, 1410, 2310, 1750, 2130, 1400, 1380,
```

```
1580, 3030, 1280, 1940, 1390, 2315, 2240, 2350, 2140, 4850, 1870,
2610, 2720, 3100, 4420, 4530, 3430, 2550, 1670, 3070, 2020, 3180,
2970, 1690, 2750, 2170, 3715, 1950, 2580, 1810, 3010, 1350, 1720,
1800, 2840, 2330, 1060, 2160, 2030, 1880, 1520, 2500, 1290, 1470,
1890, 1730, 2220, 1840, 2670, 1200, 1408, 1620, 1430, 1630, 1310,
1760, 1820, 1220, 1980, 1130, 1170, 1510, 1240, 2488, 3510, 2490,
2540, 2120, 2040, 3040, 3240, 3130, 3770, 2790, 2800, 2530, 2450,
2520, 2770, 2000, 1780, 2210, 1420, 1660, 1970, 1270, 1460, 1500,
1930, 1330, 1740, 1370, 2090, 1230, 2441,  840, 2360, 1650, 1490,
 900,  820, 1700, 4100, 2960, 3470, 3820, 2430, 4130, 2190, 1990,
2250, 3200, 2850, 2560, 1640, 2870, 2510, 1180, 2600, 1540, 1250,
1040, 1360, 1516, 2230, 2440, 2011, 1010, 1140, 1070,  910, 1326,
3450, 2930, 2900, 3260, 2920, 2950, 3620, 1900, 1210, 3140, 2300,
1190, 2527, 2150, 2980, 1920, 1600, 1357, 1572, 4460, 3890, 3660,
3230, 3500, 3080, 3880, 2700, 2690, 2100, 2270, 1110, 1439,  998,
1714, 1610, 1550, 1020, 3220, 4760, 2890, 3530, 2400, 3600, 2480,
3170, 3640, 2370,  980, 1080, 1120, 1830,  890, 1710, 3740, 4040,
4240, 4440, 3290, 2180, 3120,  990, 2650, 3060, 1364, 2420, 3480,
4560, 3210, 3390, 3360, 2910,  950,  920, 1030, 1530, 3860, 4210,
3700, 2740, 2810, 2460, 2660, 1232,  850, 3490, 3150, 1445, 2114,
1404, 3910, 3160, 3580, 2760,  930, 3300, 5170, 4060, 3920, 3610,
2303, 1862, 1050, 3850, 3840, 1000, 2110, 2680, 2050, 2620, 3790,
2415, 3440, 2640, 3110, 2052, 2095, 3630, 2710, 3270, 5030, 3680,
 970, 1571, 1307, 1658, 3540, 4290, 2358, 3370, 1665, 3494, 2434,
 860,  880, 3930, 3710, 4140, 1365, 4020, 3690, 3750, 3590, 1346,
3330, 2630, 1518, 3190, 1495, 2305, 3730, 2037, 2363, 1765, 3810,
4090, 3280, 4390, 2027,  960, 2437,  770,  700, 4900, 3960, 3050,
2578, 1484, 2583, 1914, 4280, 2412, 4070, 3380, 1405, 1811, 3250,
3550, 2518, 3020, 2106, 2009, 1188, 4630, 3800, 4670, 3950, 1295,
2478,  740, 3310, 4180, 2683, 2955, 4000, 3400, 3900, 3670, 3780,
4400, 3420,  830,  460, 1256, 1494, 1098, 3720, 3560, 2028, 1459,
1584, 3340, 2496, 1934, 2456, 4470, 4170, 3980, 1798, 2376, 2594,
2214, 1768, 4550, 4010, 2554, 4950, 1277, 1156,  940, 2667, 5080,
5790, 3830, 3639, 1664, 1481, 4080, 2502, 4620, 3410, 3090, 3618,
2912, 2238, 1078, 5070, 3970, 4490, 3570, 2516,  780, 1767, 4160,
3760, 3520, 2566, 1678, 4920, 3650, 4510, 4030, 3625, 2165, 2156,
2641, 3460, 4340,  800, 4680, 4300, 2234,  760, 3990, 4640, 1746,
1569, 1696, 2815, 1309,  870, 2458, 4750, 3045, 1894, 2648, 1802,
2598, 2154, 2029, 1616, 2738, 2634, 2166, 2673, 1137, 4270, 4310,
1979, 1537, 1847, 4150, 2996, 1546, 1813, 2704, 5380, 3721, 4190,
2475,  790, 4362,  806, 4330, 2597, 1522, 1466, 1264, 2616, 1536,
4042, 4230, 2198, 2575, 4890, 3112, 1745, 1448, 2574, 2439, 1076,
 810, 4913, 2798, 2189, 1528, 3940, 2533, 2622, 5200, 2056, 1458,
1509, 2382, 1975, 4120, 4110, 4590, 4690, 2451, 1984, 2323, 1358,
5600, 2142, 3191, 1336, 4320, 4830, 4225, 2474, 3425, 2316, 2688,
2112, 3557, 5110, 1716, 2725, 2396, 1981, 4930, 3008, 1554, 1442,
1463, 4480, 1638, 3236, 1138, 2876, 3193,  750, 2424, 2901, 4540,
1303, 1919, 2049, 2077, 1381,  710, 1282, 2612, 1941, 2136, 4370,
2875, 2555, 2304, 1443, 3159, 2767, 4940, 4570, 2425, 1268, 1399,
1356, 2221,  720, 4770, 2665, 3078, 2344, 2246, 1639, 2724, 2092,
2389, 2406, 1566, 1168,  670, 2419, 2014, 2879, 2015, 3543, 2619,
1092, 1608, 1884, 1691, 2927, 4800, 2495, 1845, 1763, 4410, 2873,
2258, 1427,  690,  620, 2405, 4200, 1415, 2547, 3087, 2091, 4650,
```

```
df.corr()
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	con
id	1.000000	0.045966	-0.329034	-0.516909	-0.648127	-0.100269	-0.312305	-0.112937	-0.293004	-0.(
Date	0.045966	1.000000	-0.015663	-0.026485	-0.021958	0.004392	-0.010335	0.012006	-0.004782	-0.(
number of bedrooms	-0.329034	-0.015663	1.000000	0.509784	0.570526	0.034416	0.177294	-0.006257	0.078665	0.(
number of bathrooms	-0.516909	-0.026485	0.509784	1.000000	0.753517	0.080806	0.502924	0.060104	0.183789	-0.1
living area	-0.648127	-0.021958	0.570526	0.753517	1.000000	0.174420	0.354743	0.105837	0.287728	-0.(
lot area	-0.100269	0.004392	0.034416	0.080806	0.174420	1.000000	-0.004138	0.026282	0.078308	-0.(
number of floors	-0.312305	-0.010335	0.177294	0.502924	0.354743	-0.004138	1.000000	0.016316	0.020153	-0.2
waterfront present	-0.112937	0.012006	-0.006257	0.060104	0.105837	0.026282	0.016316	1.000000	0.400206	0.(
number of views	-0.293004	-0.004782	0.078665	0.183789	0.287728	0.078308	0.020153	0.400206	1.000000	0.(
condition of the house	-0.045061	-0.027402	0.026597	-0.128232	-0.063358	-0.008548	-0.269928	0.018644	0.052533	1.(
grade of the house	-0.673448	-0.033097	0.352945	0.663054	0.761835	0.110546	0.463082	0.079831	0.254532	-0.1
Area of the house(excluding basement)	-0.565116	-0.015994	0.473599	0.684391	0.875793	0.183553	0.525643	0.071865	0.162672	-0.1

```
#mode
df['id'].mode()
```

```
0        6762810020
1        6762810021
2        6762810022
3        6762810023
4        6762810026
            ...
14615    6762831611
14616    6762831612
14617    6762831613
14618    6762831615
14619    6762831616
Name: id, Length: 14620, dtype: int64
```

schools nearby	-0.004821	-0.004071	0.003397	0.002180	0.002370	-0.012671	-0.007579	0.001563	0.008004	-0.(

```
#mean
df['lot area'].mean()
```

```
15093.281121751026
```

```
#median
df['lot area'].median()
```

```
7620.0
```

```
round(df['lot area'].std(),2)
```

```
37919.62
```

```
round(df.describe(),2)
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house
count	1.462000e+04	14620.00	14620.00	14620.00	14620.00	14620.00	14620.00	14620.00	14620.00	14620.00
mean	6.762821e+09	42604.54	3.38	2.13	2098.26	15093.28	1.50	0.01	0.23	3.43
std	6.237570e+03	67.35	0.94	0.77	928.28	37919.62	0.54	0.09	0.77	0.66
min	6.762810e+09	42491.00	1.00	0.50	370.00	520.00	1.00	0.00	0.00	1.00
25%	6.762815e+09	42546.00	3.00	1.75	1440.00	5010.75	1.00	0.00	0.00	3.00
50%	6.762821e+09	42600.00	3.00	2.25	1930.00	7620.00	1.50	0.00	0.00	3.00
75%	6.762826e+09	42662.00	4.00	2.50	2570.00	10800.00	2.00	0.00	0.00	4.00

```
#statistics by groups
df['lot area'].groupby(df['id'])
```

```
<pandas.core.groupby.generic.SeriesGroupBy object at 0x7f89d72697f0>
```

Handle the missing values

```
#replacing with arbitary value
df.fillna(method='ffill')
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	
...	...	...	...	...	...	...	...	...	...	...	...	...	
14615	6762830250	42734	2	1.50	1556	20000	1.0	0	0	4	...	1957	
14616	6762830339	42734	3	2.00	1680	7000	1.5	0	0	4	...	1968	
14617	6762830618	42734	2	1.00	1070	6120	1.0	0	0	3	...	1962	
14618	6762830709	42734	4	1.00	1030	6621	1.0	0	0	4	...	1955	
14619	6762831463	42734	3	1.00	900	4770	1.0	0	0	3	...	1969	

14620 rows × 23 columns

```
# checking the missing value
df.isnull().sum()
```

```
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
dtype: int64
```

```
#filling missing values with 0
new_df=df.fillna(0)
new_df
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present
0	6762810145	42491	5	2.50	3650	9050	2.0	0
1	6762810635	42491	4	2.50	2920	4000	1.5	0
2	6762810998	42491	5	2.75	2910	9480	1.5	0
3	6762812605	42491	4	2.50	3310	42998	2.0	0
4	6762812919	42491	3	2.00	2710	4500	1.5	0
...	...	...	...	...	...	...	...	...
14615	6762830250	42734	2	1.50	1556	20000	1.0	0
14616	6762830339	42734	3	2.00	1680	7000	1.5	0
14617	6762830618	42734	2	1.00	1070	6120	1.0	0
14618	6762830709	42734	4	1.00	1030	6621	1.0	0
14619	6762831463	42734	3	1.00	900	4770	1.0	0

14620 rows × 23 columns

```
#filling NaN values with forward fill value
new_df=df.fillna(method="ffill")
new_df
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	

```
#interpolate of missing values
new_df=df.interpolate()
df
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	
...	...	...	...	...	...	...	...	...	...	...	...	...	
14615	6762830250	42734	2	1.50	1556	20000	1.0	0	0	4	...	1957	
14616	6762830339	42734	3	2.00	1680	7000	1.5	0	0	4	...	1968	
14617	6762830618	42734	2	1.00	1070	6120	1.0	0	0	3	...	1962	
14618	6762830709	42734	4	1.00	1030	6621	1.0	0	0	4	...	1955	
14619	6762831463	42734	3	1.00	900	4770	1.0	0	0	3	...	1969	

14620 rows × 23 columns

```
#dropna()
new_df=df.dropna()
new_df
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	
...	...	...	...	...	...	...	...	...	...	...	...	...	
14615	6762830250	42734	2	1.50	1556	20000	1.0	0	0	4	...	1957	
14616	6762830339	42734	3	2.00	1680	7000	1.5	0	0	4	...	1968	
14617	6762830618	42734	2	1.00	1070	6120	1.0	0	0	3	...	1962	
14618	6762830709	42734	4	1.00	1030	6621	1.0	0	0	4	...	1955	
14619	6762831463	42734	3	1.00	900	4770	1.0	0	0	3	...	1969	

14620 rows × 23 columns

```
#deleting the rows  having all NaN values
new_df=df.dropna(how='all')
new_df
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Re
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	
...	...	...	...	...	...	...	...	...	...	...	...	...	
14615	6762830250	42734	2	1.50	1556	20000	1.0	0	0	4	...	1957	
14616	6762830339	42734	3	2.00	1680	7000	1.5	0	0	4	...	1968	
14617	6762830618	42734	2	1.00	1070	6120	1.0	0	0	3	...	1962	
14618	6762830709	42734	4	1.00	1030	6621	1.0	0	0	4	...	1955	
14619	6762831463	42734	3	1.00	900	4770	1.0	0	0	3	...	1969	

14620 rows × 23 columns