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

```
hf = pd.read_csv("/content/House Price India.csv")
```

```
hf.head()
```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	cc
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

```
hf.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	Renovation Year	Postal Code	Latt
14615	6762830250	42734	2	1.5	1556	20000	1.0	0	0	4	...	1957	0	122066	5
14616	6762830339	42734	3	2.0	1680	7000	1.5	0	0	4	...	1968	0	122072	5
14617	6762830618	42734	2	1.0	1070	6120	1.0	0	0	3	...	1962	0	122056	5
14618	6762830709	42734	4	1.0	1030	6621	1.0	0	0	4	...	1955	0	122042	5
14619	6762831463	42734	3	1.0	900	4770	1.0	0	0	3	...	1969	2009	122018	5

5 rows × 23 columns

```
hf.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
```

```
hf.isnull()
```

	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	Renovation Year	Postal Code	Lattitude
0	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
14615	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
14616	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
14617	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
14618	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False
14619	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False

14620 rows × 23 columns

```
hf.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
```

```
print(hf.describe())
```

```
       [illegible]  [illegible]  [illegible]  [illegible]  [illegible]
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     292.075376  1.509328e+04          1.502360            0.007661
std      179.469511  3.791962e+04          0.540239            0.087193
min        0.000000  5.200000e+02          1.000000            0.000000
25%      146.000000  5.010750e+03          1.000000            0.000000
50%      259.000000  7.620000e+03          1.500000            0.000000
75%      414.000000  1.080000e+04          2.000000            0.000000
max      864.000000  1.074218e+06          3.500000            1.000000
```

```
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]
```

```
from sklearn.preprocessing import LabelEncoder
```

```
le = LabelEncoder()
```

```
hf["living area"]=le.fit_transform(hf["living area"])
```

```
hf.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	Renovation Year	Postal Code	Lattituc
0	6762810145	42491	5	2.50	602	9050	2.0	0	4	5	...	1921	0	122003	52.864
1	6762810635	42491	4	2.50	493	4000	1.5	0	0	5	...	1909	0	122004	52.887
2	6762810998	42491	5	2.75	492	9480	1.5	0	0	3	...	1939	0	122004	52.885
3	6762812605	42491	4	2.50	559	42998	2.0	0	0	3	...	2001	0	122005	52.953
4	6762812919	42491	3	2.00	449	4500	1.5	0	0	4	...	1929	0	122006	52.904

5 rows × 23 columns

```
hf.dtypes
```

```
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
```

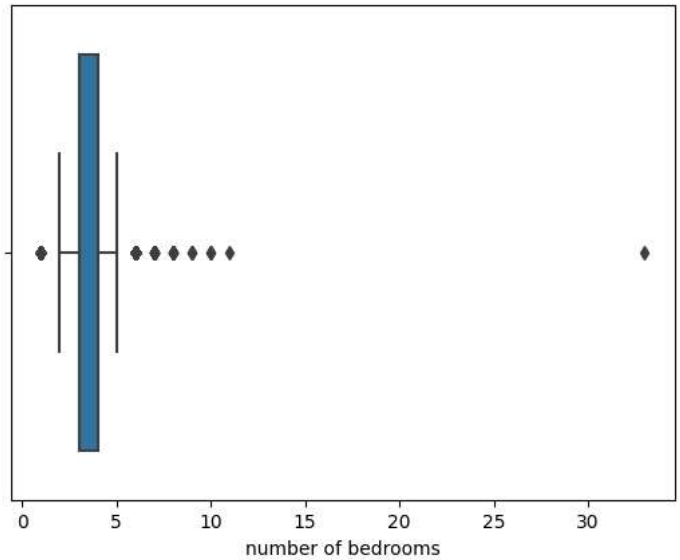
```
hf.columns
```

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

```
hf.describe()
```

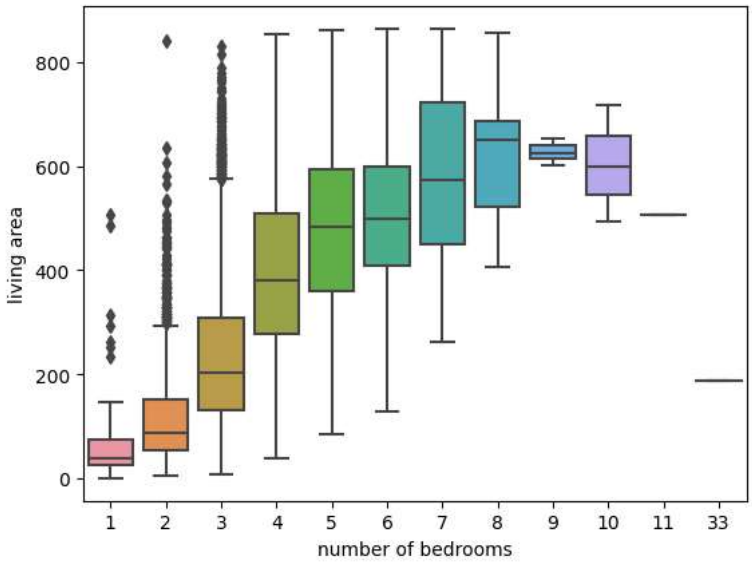
```
sns.boxplot(x=hf['number of bedrooms'])
```

```
<Axes: xlabel='number of bedrooms'>
```



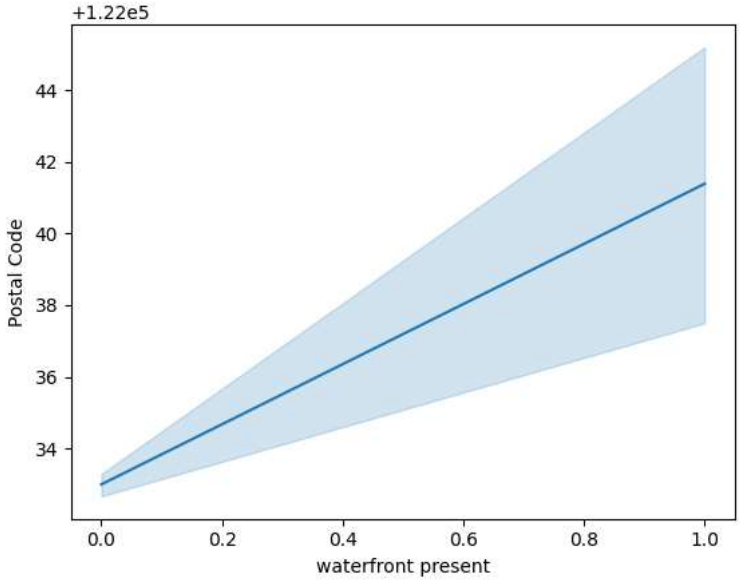
```
sns.boxplot(x=hf['number of bedrooms'],y=hf['living area'])
```

```
<Axes: xlabel='number of bedrooms', ylabel='living area'>
```




```
sns.lineplot(x=hf['waterfront present'],y=hf['Postal Code'])
```

```
<Axes: xlabel='waterfront present', ylabel='Postal Code'>
```

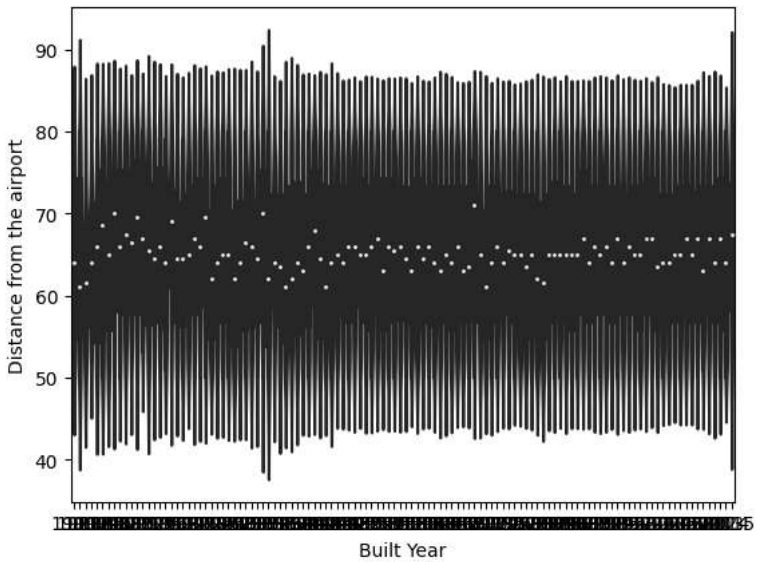



```
plt.hist(hf['condition of the house'],bins=50)
```

```
(array([  18.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,
          0.,    0.,    0.,  100.,    0.,    0.,    0.,    0.,    0.,
          0.,    0.,    0.,    0.,    0.,    0.,    0., 9350.,    0.,
          0.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,
          0., 3874.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,
          0.,    0.,    0.,    0., 1278.]),
 array([1.  , 1.08, 1.16, 1.24, 1.32, 1.4 , 1.48, 1.56, 1.64, 1.72, 1.8 ,
        1.88, 1.96, 2.04, 2.12, 2.2 , 2.28, 2.36, 2.44, 2.52, 2.6 , 2.68,
        2.76, 2.84, 2.92, 3.  , 3.08, 3.16, 3.24, 3.32, 3.4 , 3.48, 3.56,
        3.64, 3.72, 3.8 , 3.88, 3.96, 4.04, 4.12, 4.2 , 4.28, 4.36, 4.44,
        4.52, 4.6 , 4.68, 4.76, 4.84, 4.92, 5.  ]),
 <BarContainer object of 50 artists>)
```

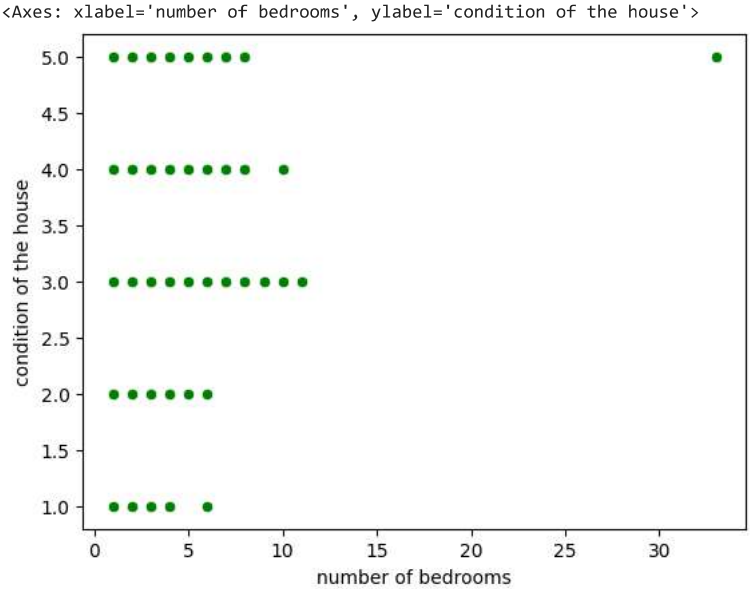
```
sns.violinplot(x=hf['Built Year'],y=hf['Distance from the airport'],color='purple')
```

```
<Axes: xlabel='Built Year', ylabel='Distance from the airport'>
```



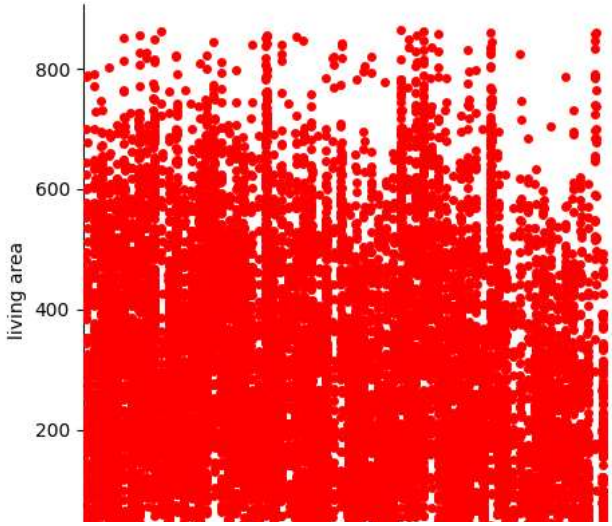

```
sns.scatterplot(x=hf['number of bedrooms'],y=hf['condition of the house'],color='g')
```

```
<Axes: xlabel='number of bedrooms', ylabel='condition of the house'>
```




```
sns.catplot(x=hf['Postal Code'],y=hf['living area'],color='r')
```

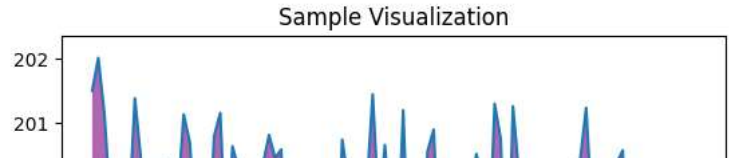
```
<seaborn.axisgrid.FacetGrid at 0x785a7343ef50>
```




```
sns.barplot(x=hf['Price'],y=hf['Longitude'],color='y')
```

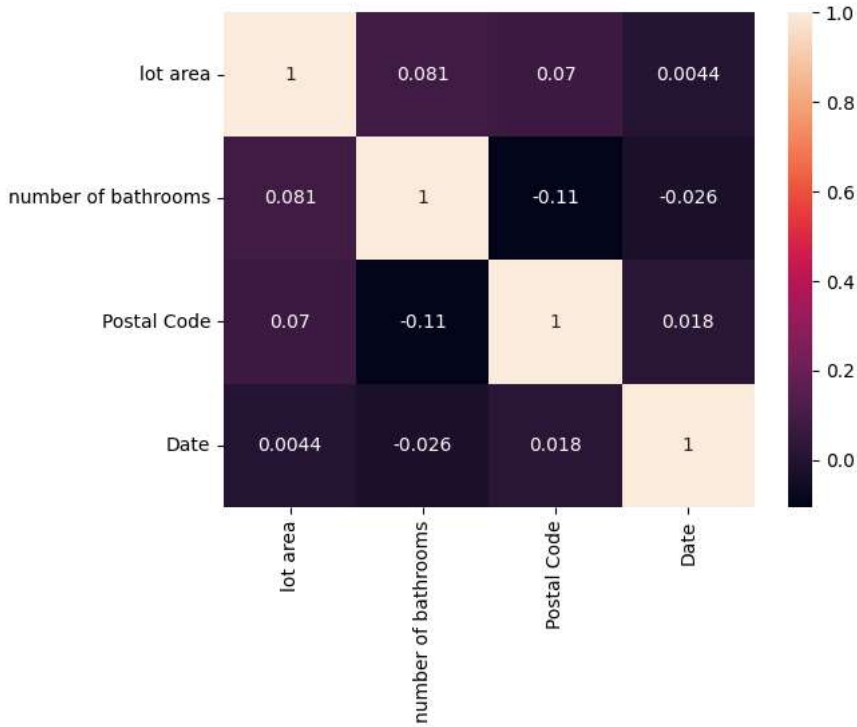
```
<Axes: xlabel='Price', ylabel='Longitude'>
```


```
ys = 200 + np.random.randn(100)
x = [x for x in range(len(ys))]

plt.plot(x, ys, '-')
plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='purple', alpha=0.6)

plt.title("Sample Visualization")
plt.show()
```

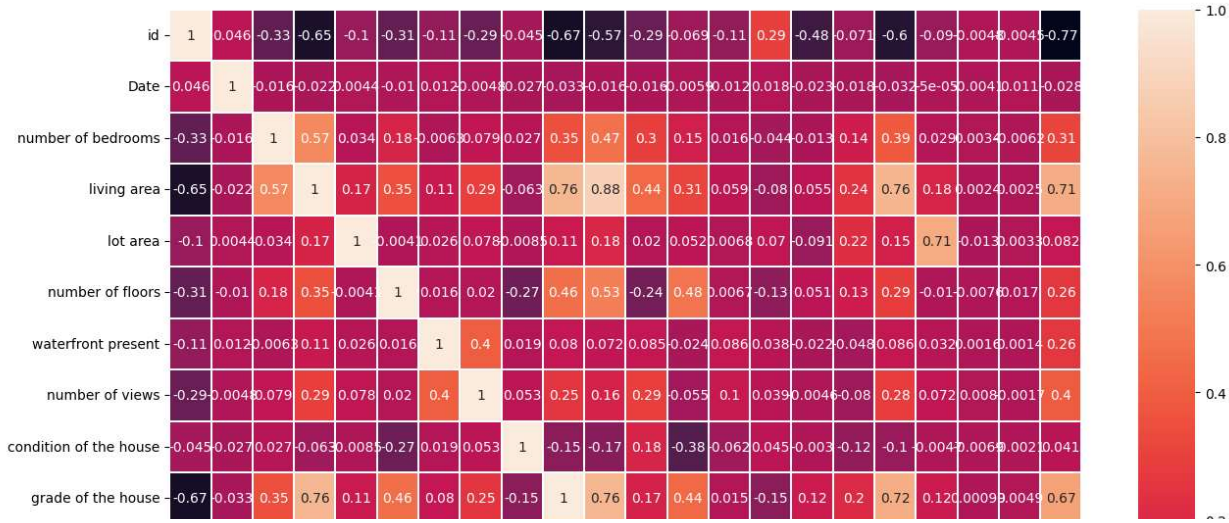



```
sns.heatmap(hf[['lot area','number of bathrooms','Postal Code','Date']].corr(),annot=True,color='r')
```

```
<Axes: >
```




```
plt.subplots(figsize=(15,15))
sns.heatmap(df.drop(['number of bathrooms'],axis=1).corr(),linewidth=0.3,annot=True)
plt.show()
```




```
print(hf.describe())
```

```
       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     292.075376  1.509328e+04          1.502360            0.007661
       std      179.469511  3.791962e+04          0.540239            0.087193
       min        0.000000  5.200000e+02          1.000000            0.000000
       25%      146.000000  5.010750e+03          1.000000            0.000000
       50%      259.000000  7.620000e+03          1.500000            0.000000
       75%      414.000000  1.080000e+04          2.000000            0.000000
       max      864.000000  1.074218e+06          3.500000            1.000000

              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
```

```
[8 rows x 23 columns]
```

```
print(hf.count())
```

```
id                                      14620
Date                                    14620
number of bedrooms                      14620
number of bathrooms                     14620
living area                             14620
lot area                                14620
number of floors                        14620
waterfront present                      14620
number of views                         14620
condition of the house                  14620
grade of the house                      14620
Area of the house(excluding basement)   14620
Area of the basement                    14620
Built Year                              14620
Renovation Year                         14620
Postal Code                             14620
Lattitude                               14620
Longitude                               14620
living_area_renov                       14620
lot_area_renov                          14620
Number of schools nearby                14620
Distance from the airport               14620
Price                                   14620
dtype: int64
```

```
hf.dropna(inplace=True)
hf.fillna(0,inplace=True)
hf.interpolate(inplace=True)
```

```
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
x=hf.drop(['lot area','Date'],axis=1)
x.set_index(['id'],inplace=True)
y=hf[['id','Date']]
y.head()
```

	id	Date
0	6762810145	42491
1	6762810635	42491
2	6762810998	42491
3	6762812605	42491
4	6762812919	42491

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import r2_score
```

```
print('Mean:',hf['Number of schools nearby'].mean())
print('Median:',hf['Area of the house(excluding basement)'].median())
print('Mode:',hf['grade of the house'].mode())
```

```
Mean: 2.0122435020519838
Median: 1580.0
Mode: 0    7
Name: grade of the house, dtype: int64
```

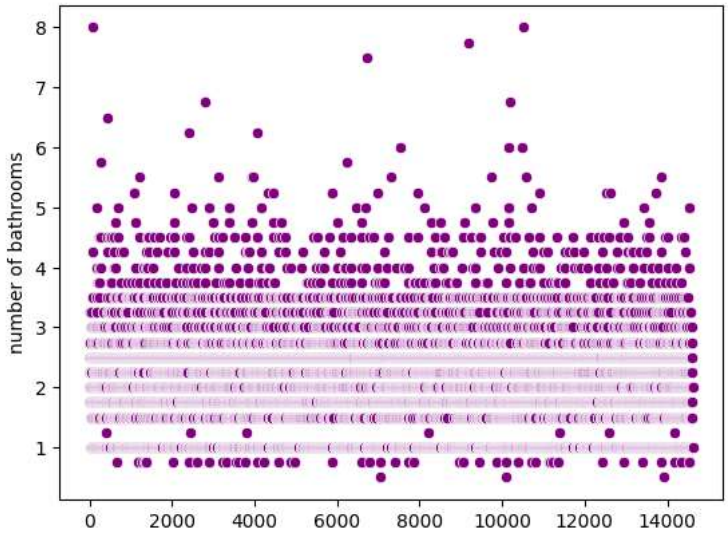
```
print(hf.isnull().sum())
```

```
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
```

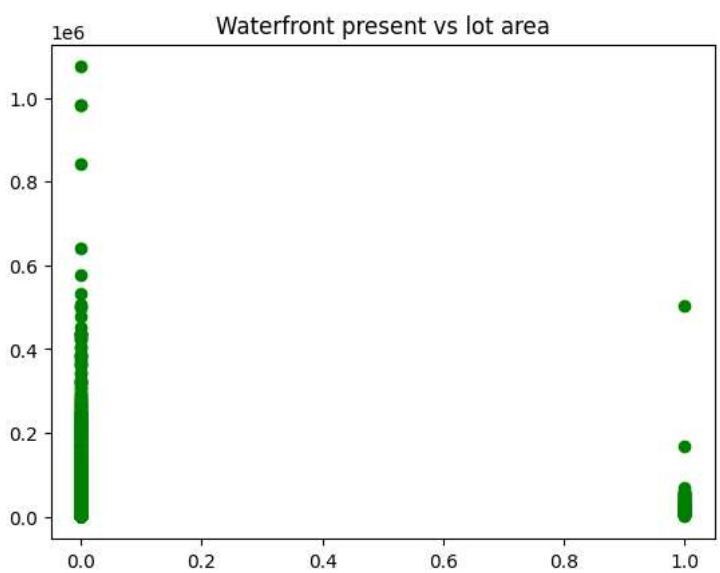
```
Area of the house(excluding basement)    0
Area of the basement                     0
Built Year                               0
Renovation Year                          0
Postal Code                              0
Lattitude                                0
Longitude                                0
living_area_renov                        0
lot_area_renov                           0
Number of schools nearby                 0
Distance from the airport                0
Price                                    0
dtype: int64
```

```
sns.scatterplot(hf['number of bathrooms'],color='purple')
```

```
<Axes: ylabel='number of bathrooms'>
```




```
plt.scatter(hf['waterfront present'],hf['lot area'],color='g')
plt.title("Waterfront present vs lot area")
plt.grid(linestyle='-', linewidth=0.)
```

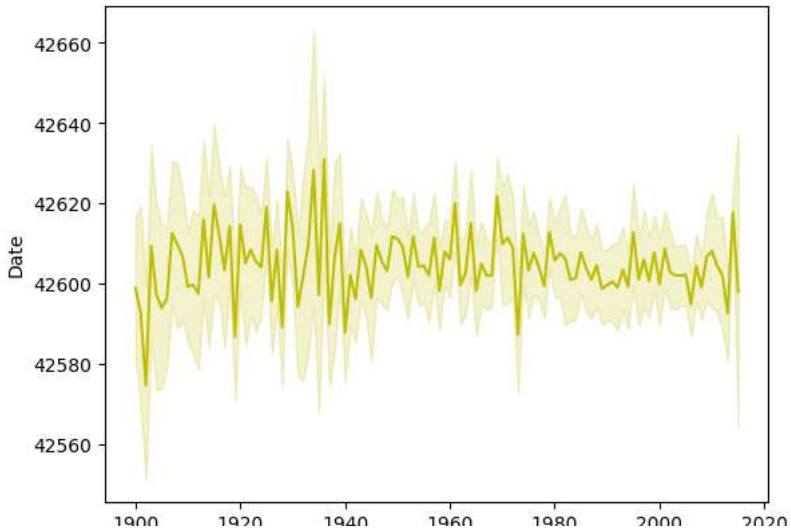



```
hf.duplicated().sum()
```

```
0
```

```
sns.lineplot(x=hf['Built Year'],y=hf['Date'],color='y')
```

```
<Axes: xlabel='Built Year', ylabel='Date'>
```




```
sns.jointplot(data =hf,x= 'lot area',y= 'living area',color='g')
```

```
<seaborn.axisgrid.JointGrid at 0x785a75f067d0>
```