In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
```

```
C:\Users\DELL\anaconda3\lib\site-packages\scipy\__init__.py:146: UserWarning: A NumPy vers
ion >=1.16.5 and <1.23.0 is required for this version of SciPy (detected version 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

In [2]:
```
df = pd.read_csv('kc_house_data.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	...	grad
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650	1.0	0	0	...	
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242	2.0	0	0	...	
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000	1.0	0	0	...	
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000	1.0	0	0	...	
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080	1.0	0	0	...	

5 rows × 21 columns

In [4]:
```
# Count the null values
df.isnull().sum() # đếm số lượng giá trị null trong bảng
```

Out[4]:
```
id               0
date             0
price            0
bedrooms         0
bathrooms        0
sqft_living      0
sqft_lot         0
floors           0
waterfront       0
view             0
condition        0
grade            0
sqft_above       0
sqft_basement    0
yr_built         0
yr_renovated     0
zipcode          0
lat              0
long             0
sqft_living15    0
sqft_lot15       0
dtype: int64
```

In [5]:
```
# describe and validate parameters between variables
df.describe().transpose() # mô tả + đánh giá độ lệch chuẩn trung bình giữa các dữ liệu
```

Out[5]:

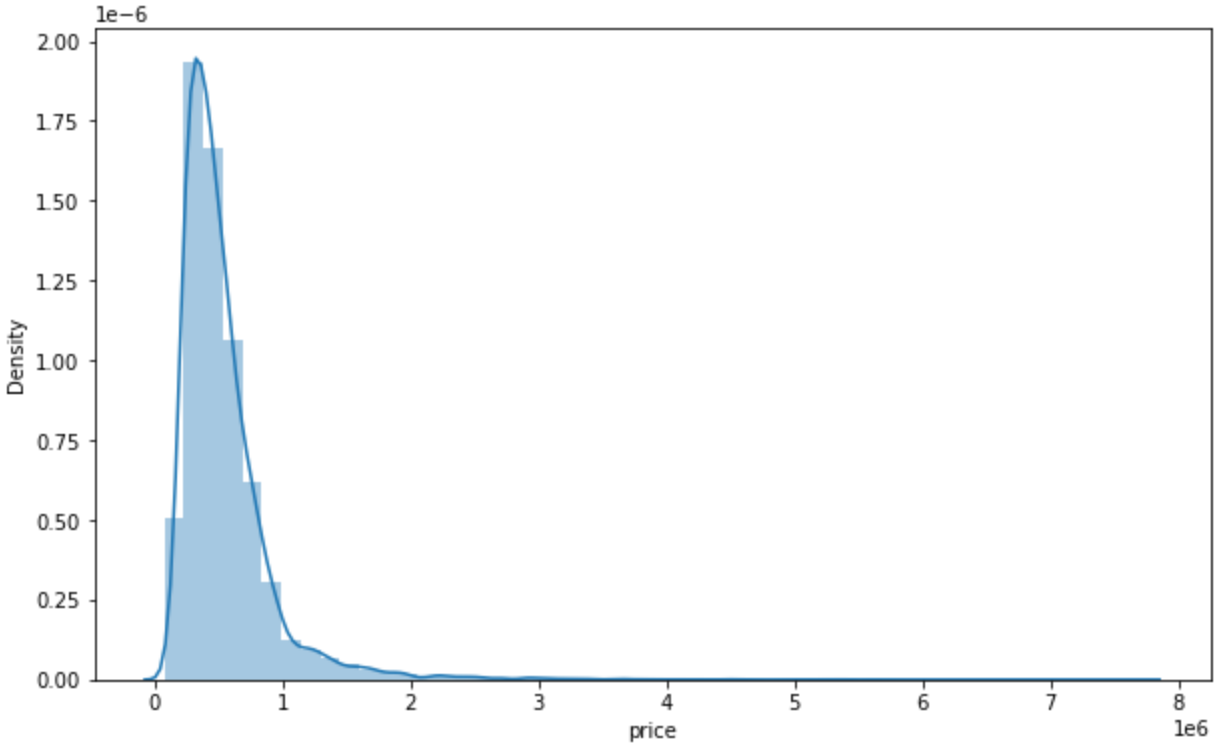
	count	mean	std	min	25%	50%	75%
id	21597.0	4.580474e+09	2.876736e+09	1.000102e+06	2.123049e+09	3.904930e+09	7.308900e+09

	count	mean	std	min	25%	50%	75%
price	21597.0	5.402966e+05	3.673681e+05	7.800000e+04	3.220000e+05	4.500000e+05	6.450000e+05
bedrooms	21597.0	3.373200e+00	9.262989e-01	1.000000e+00	3.000000e+00	3.000000e+00	4.000000e+00
bathrooms	21597.0	2.115826e+00	7.689843e-01	5.000000e-01	1.750000e+00	2.250000e+00	2.500000e+00
sqft_living	21597.0	2.080322e+03	9.181061e+02	3.700000e+02	1.430000e+03	1.910000e+03	2.550000e+03
sqft_lot	21597.0	1.509941e+04	4.141264e+04	5.200000e+02	5.040000e+03	7.618000e+03	1.068500e+04
floors	21597.0	1.494096e+00	5.396828e-01	1.000000e+00	1.000000e+00	1.500000e+00	2.000000e+00
waterfront	21597.0	7.547345e-03	8.654900e-02	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
view	21597.0	2.342918e-01	7.663898e-01	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
condition	21597.0	3.409825e+00	6.505456e-01	1.000000e+00	3.000000e+00	3.000000e+00	4.000000e+00
grade	21597.0	7.657915e+00	1.173200e+00	3.000000e+00	7.000000e+00	7.000000e+00	8.000000e+00
sqft_above	21597.0	1.788597e+03	8.277598e+02	3.700000e+02	1.190000e+03	1.560000e+03	2.210000e+03
sqft_basement	21597.0	2.917250e+02	4.426678e+02	0.000000e+00	0.000000e+00	0.000000e+00	5.600000e+02
yr_built	21597.0	1.971000e+03	2.937523e+01	1.900000e+03	1.951000e+03	1.975000e+03	1.997000e+03
yr_renovated	21597.0	8.446479e+01	4.018214e+02	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
zipcode	21597.0	9.807795e+04	5.351307e+01	9.800100e+04	9.803300e+04	9.806500e+04	9.811800e+04
lat	21597.0	4.756009e+01	1.385518e-01	4.715590e+01	4.747110e+01	4.757180e+01	4.767800e+01
long	21597.0	-1.222140e+02	1.407235e-01	-1.225190e+02	-1.223280e+02	-1.222310e+02	-1.221250e+02
sqft_living15	21597.0	1.986620e+03	6.852305e+02	3.990000e+02	1.490000e+03	1.840000e+03	2.360000e+03
sqft_lot15	21597.0	1.275828e+04	2.727444e+04	6.510000e+02	5.100000e+03	7.620000e+03	1.008300e+04

In [6]:
```
plt.figure(figsize=(10,6))
sns.distplot(df['price'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `d
istplot` is a deprecated function and will be removed in a future version. Please adapt yo
ur code to use either `displot` (a figure-level function with similar flexibility) or `his
tplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

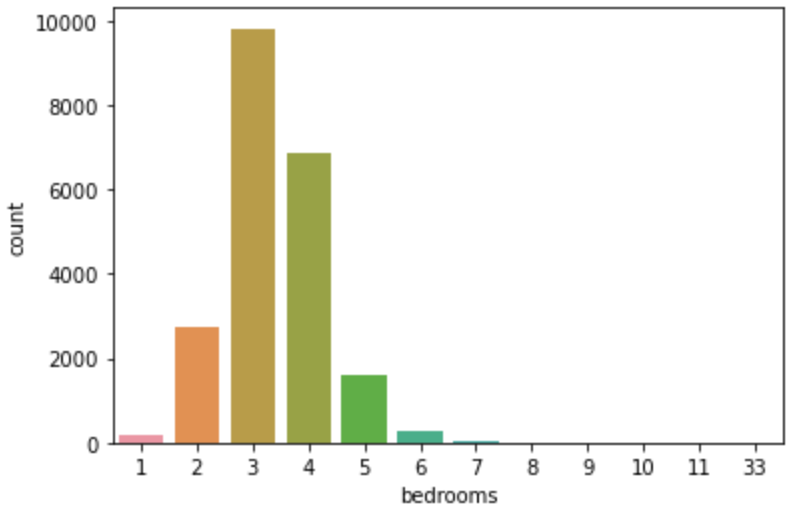
Out[6]:
```
<AxesSubplot:xlabel='price', ylabel='Density'>
```



In [7]:
```
sns.countplot(df['bedrooms'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass t
he following variable as a keyword arg: x. From version 0.12, the only valid positional ar
gument will be `data`, and passing other arguments without an explicit keyword will result
in an error or misinterpretation.
  warnings.warn(
```

Out[7]:
```
<AxesSubplot:xlabel='bedrooms', ylabel='count'>
```



In [8]:
```
df.corr()['price'].sort_values()
```

Out[8]:
```
zipcode         -0.053402
id              -0.016772
long             0.022036
condition        0.036056
yr_built         0.053953
sqft_lot15       0.082845
sqft_lot         0.089876
yr_renovated     0.126424
floors           0.256804
waterfront       0.266398
```

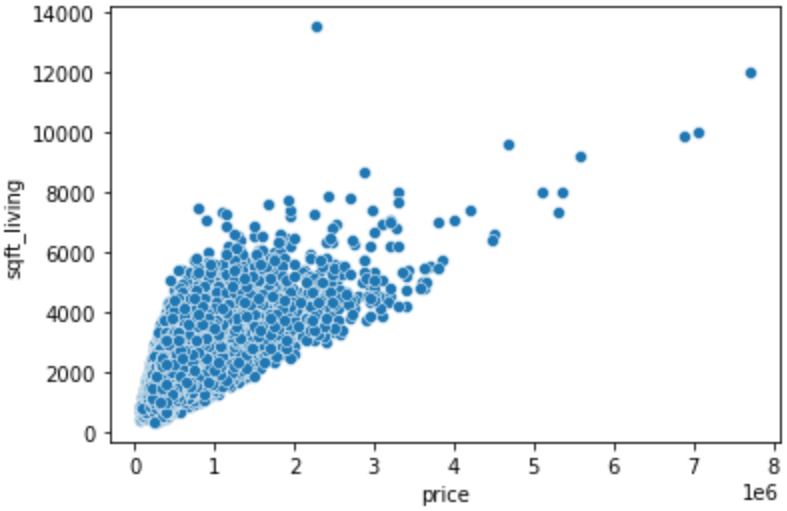
```
lat              0.306692
bedrooms         0.308787
sqft_basement    0.323799
view             0.397370
bathrooms        0.525906
sqft_living15    0.585241
sqft_above       0.605368
grade            0.667951
sqft_living      0.701917
price            1.000000
Name: price, dtype: float64
```

In [9]:

```
# plot the relationship between price and sqft_living
sns.scatterplot(x='price', y='sqft_living', data=df)
```

Out[9]:

```
<AxesSubplot:xlabel='price', ylabel='sqft_living'>
```

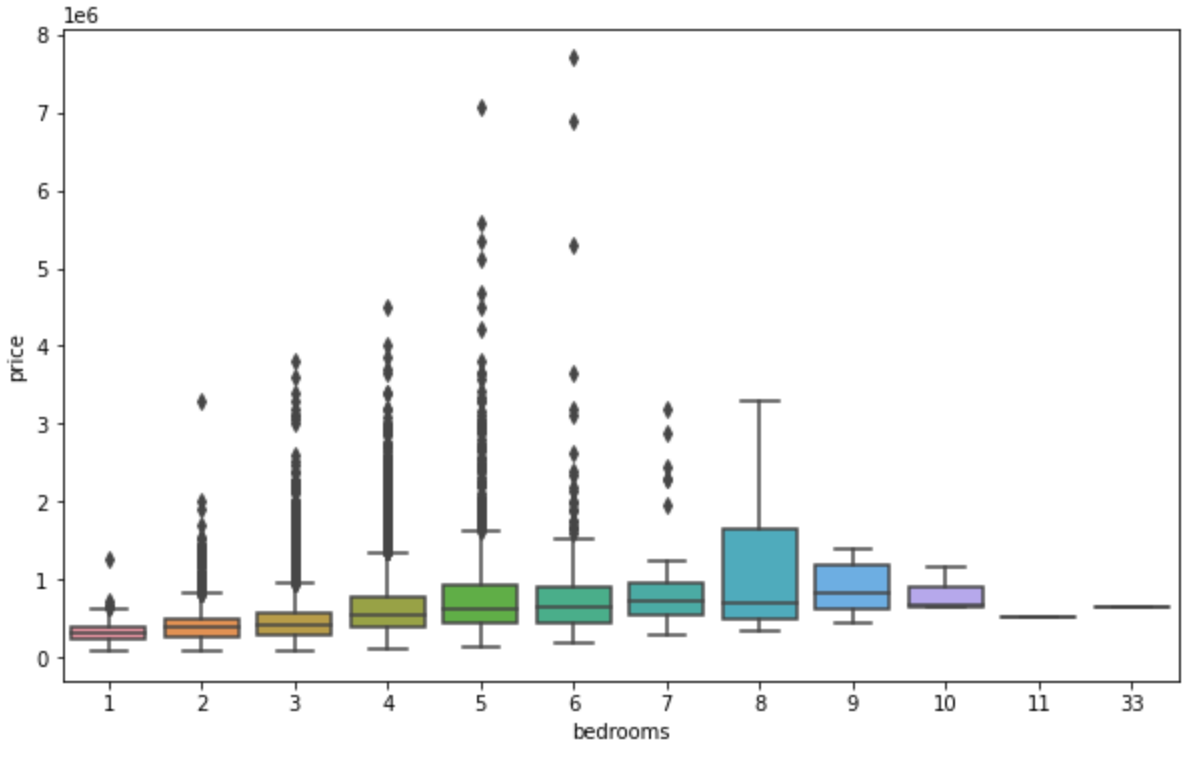


In [10]:

```
plt.figure(figsize=(10,6))
sns.boxplot(x='bedrooms',y='price',data=df)
```

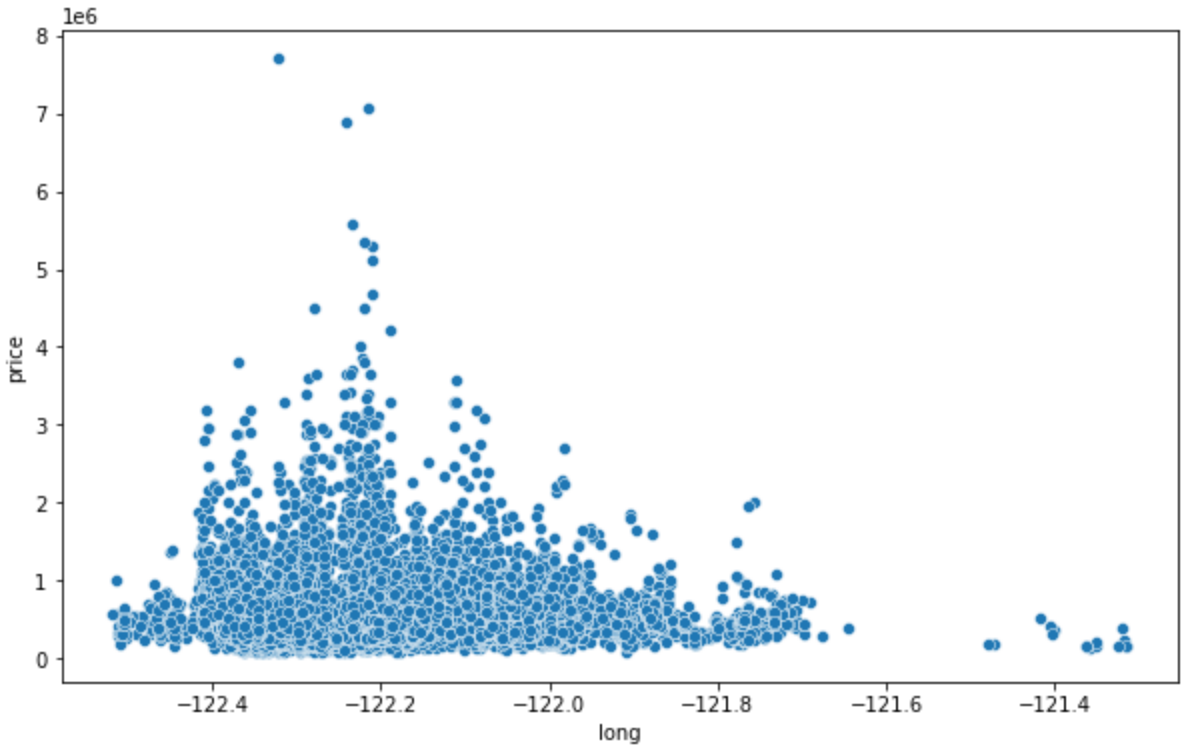
Out[10]:

```
<AxesSubplot:xlabel='bedrooms', ylabel='price'>
```



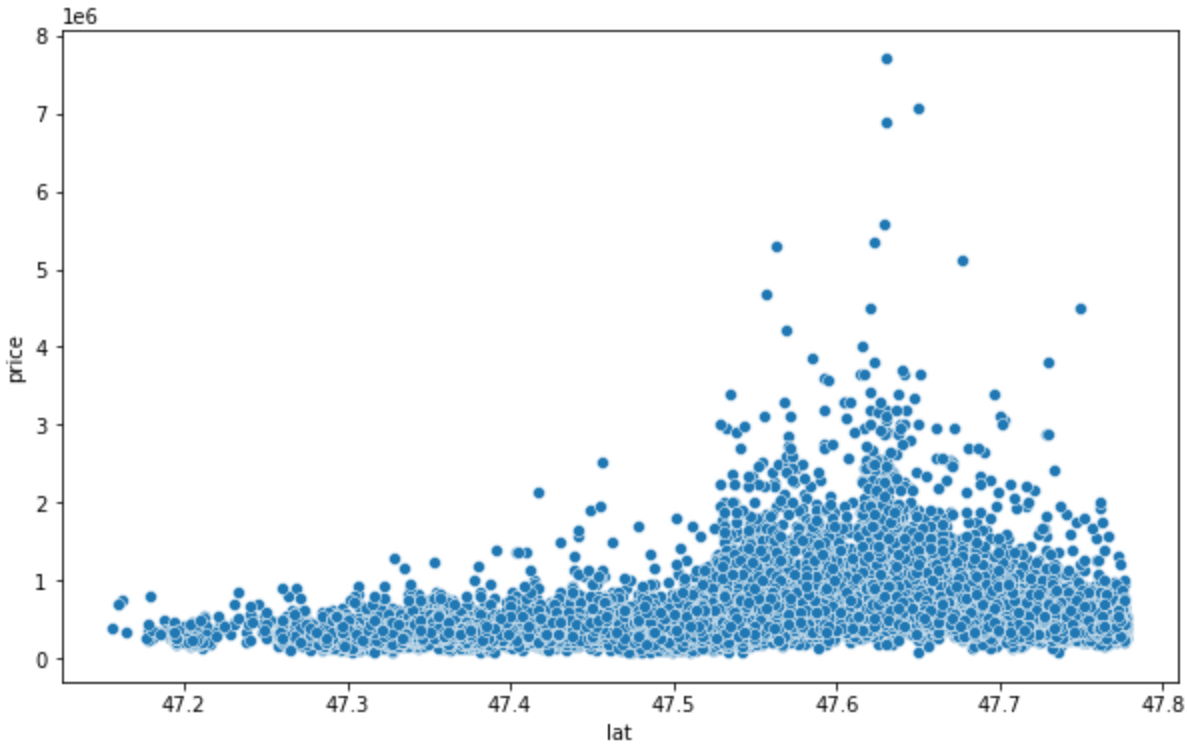
In [11]:
```
plt.figure(figsize=(10,6))
sns.scatterplot(x='long',y='price',data=df)
```

Out[11]:
```
<AxesSubplot:xlabel='long', ylabel='price'>
```



In [12]:
```
plt.figure(figsize=(10,6))
sns.scatterplot(x='lat',y='price',data=df)
```

Out[12]:
```
<AxesSubplot:xlabel='lat', ylabel='price'>
```



In [13]:
```
plt.figure(figsize=(12,6))
sns.scatterplot(x='long',y='lat',data=df, hue='price')
```

Out[13]:
```
<AxesSubplot:xlabel='long', ylabel='lat'>
```

In [14]:
```
df.sort_values('price', ascending=False).head(20)
```

Out[14]:

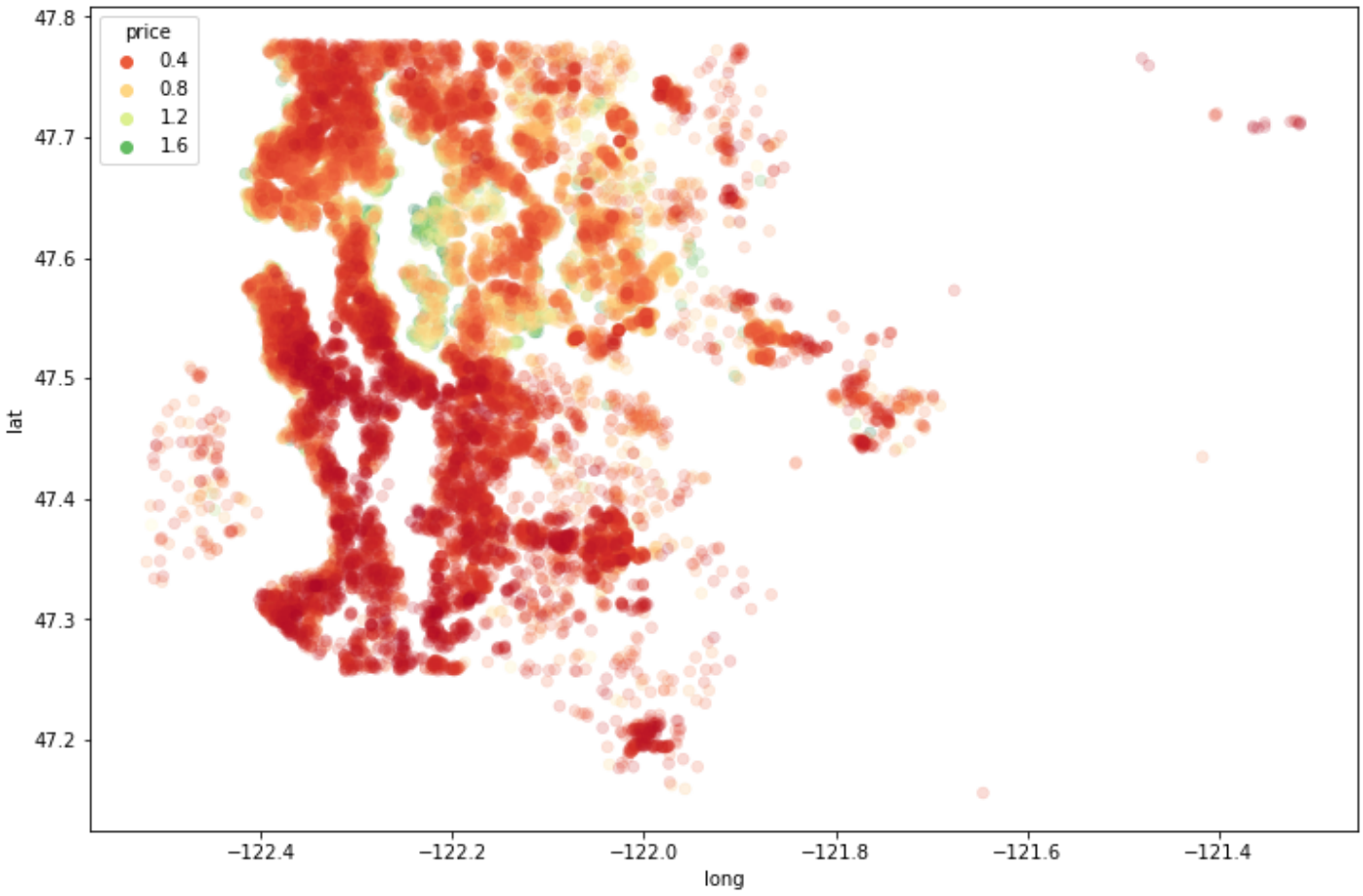
	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	...
7245	6762700020	10/13/2014	7700000.0	6	8.00	12050	27600	2.5	0	3	...
3910	9808700762	6/11/2014	7060000.0	5	4.50	10040	37325	2.0	1	2	...
9245	9208900037	9/19/2014	6890000.0	6	7.75	9890	31374	2.0	0	4	...
4407	2470100110	8/4/2014	5570000.0	5	5.75	9200	35069	2.0	0	0	...
1446	8907500070	4/13/2015	5350000.0	5	5.00	8000	23985	2.0	0	4	...
1313	7558700030	4/13/2015	5300000.0	6	6.00	7390	24829	2.0	1	4	...
1162	1247600105	10/20/2014	5110000.0	5	5.25	8010	45517	2.0	1	4	...
8085	1924059029	6/17/2014	4670000.0	5	6.75	9640	13068	1.0	1	4	...
2624	7738500731	8/15/2014	4500000.0	5	5.50	6640	40014	2.0	1	4	...
8629	3835500195	6/18/2014	4490000.0	4	3.00	6430	27517	2.0	0	0	...
12358	6065300370	5/6/2015	4210000.0	5	6.00	7440	21540	2.0	0	0	...
4145	6447300265	10/14/2014	4000000.0	4	5.50	7080	16573	2.0	0	0	...
2083	8106100105	11/14/2014	3850000.0	4	4.25	5770	21300	2.0	1	4	...
7028	853200010	7/1/2014	3800000.0	5	5.50	7050	42840	1.0	0	2	...
19002	2303900100	9/11/2014	3800000.0	3	4.25	5510	35000	2.0	0	4	...
16288	7397300170	5/30/2014	3710000.0	4	3.50	5550	28078	2.0	0	2	...
18467	4389201095	5/11/2015	3650000.0	5	3.75	5020	8694	2.0	0	1	...
6502	4217402115	4/21/2015	3650000.0	6	4.75	5480	19401	1.5	1	4	...
15241	2425049063	9/11/2014	3640000.0	4	3.25	4830	22257	2.0	1	4	...
19133	3625049042	10/11/2014	3640000.0	5	6.00	5490	19897	2.0	0	0	...

20 rows × 21 columns

In [15]:
```
#Loại bỏ những ngôi nhà đắt tiền (số liệu hiếm ảnh hưởng đến mô hình học máy)
#Exclude a few expensive houses (outliner), can be impact to the result of machine learni
non_top_1_perc = df.sort_values('price', ascending=False).iloc[216:]
```

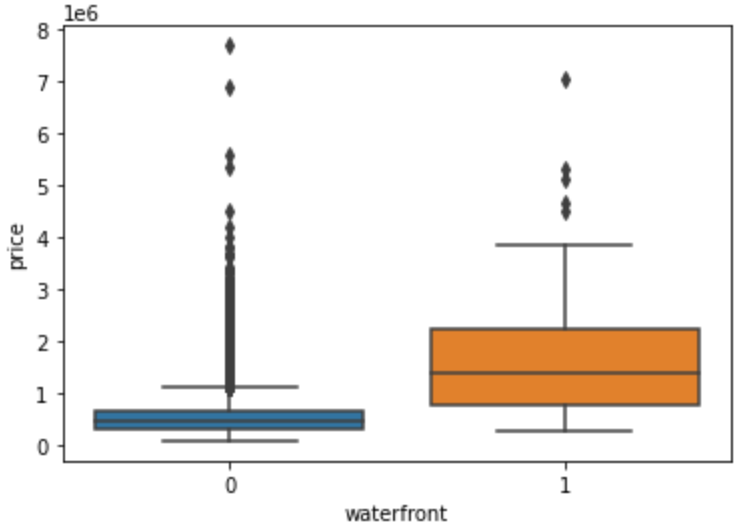
In [16]:
```
plt.figure(figsize=(12,8))
sns.scatterplot(x='long',y='lat',data=non_top_1_perc, edgecolor=None, alpha=0.2, palette='
```

Out[16]:
```
<AxesSubplot:xlabel='long', ylabel='lat'>
```



In [17]:
```
sns.boxplot(x='waterfront',y='price', data=df)
```

Out[17]:
```
<AxesSubplot:xlabel='waterfront', ylabel='price'>
```



In [19]:
```
df=df.drop('id',axis=1)
```

In [21]:
```
df['date']=pd.to_datetime(df['date'])
```

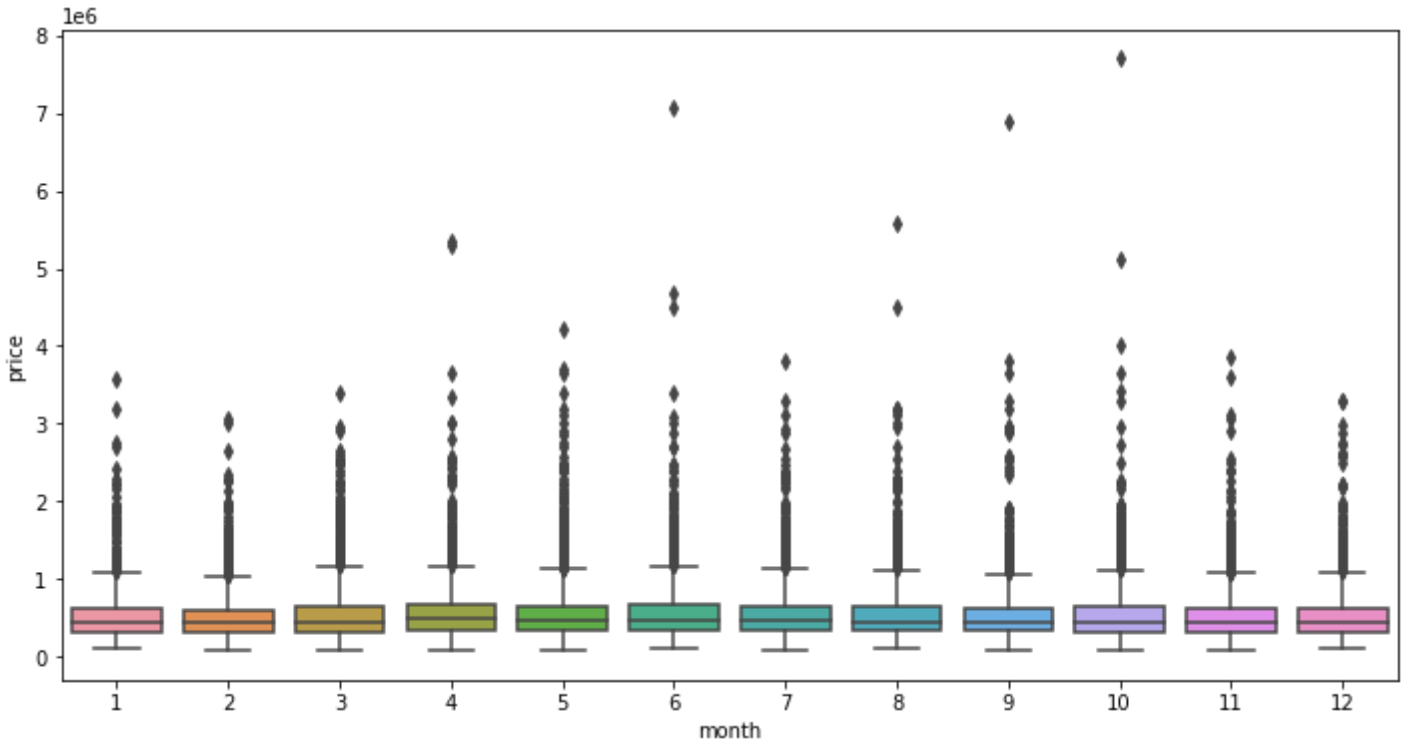
In [22]:
```
df['date']
```

Out[22]:
```
0       2014-10-13
1       2014-12-09
2       2015-02-25
3       2014-12-09
4       2015-02-18
           ...
21592   2014-05-21
21593   2015-02-23
21594   2014-06-23
21595   2015-01-16
21596   2014-10-15
Name: date, Length: 21597, dtype: datetime64[ns]
```

In [23]:
```
df['year'] = df['date'].apply(lambda date: date.year)
df['month'] = df['date'].apply(lambda date: date.month)
```

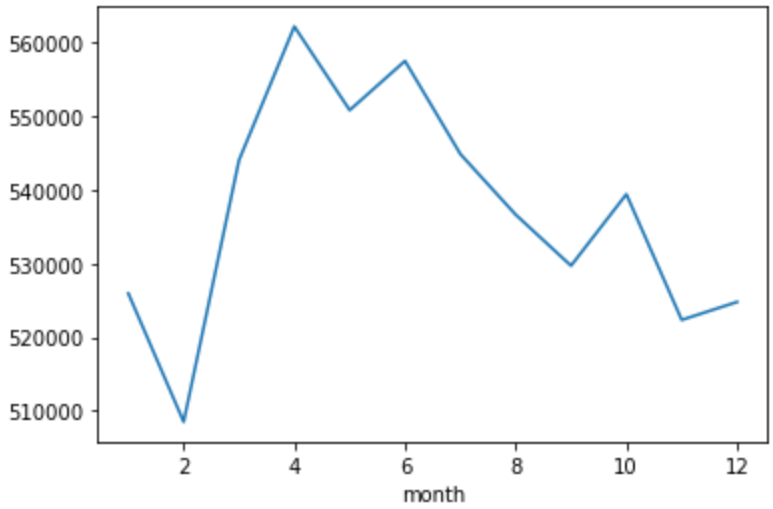
In [26]:
```
plt.figure(figsize=(12,6))
sns.boxplot(x='month', y='price', data=df)
```

Out[26]:
```
<AxesSubplot:xlabel='month', ylabel='price'>
```



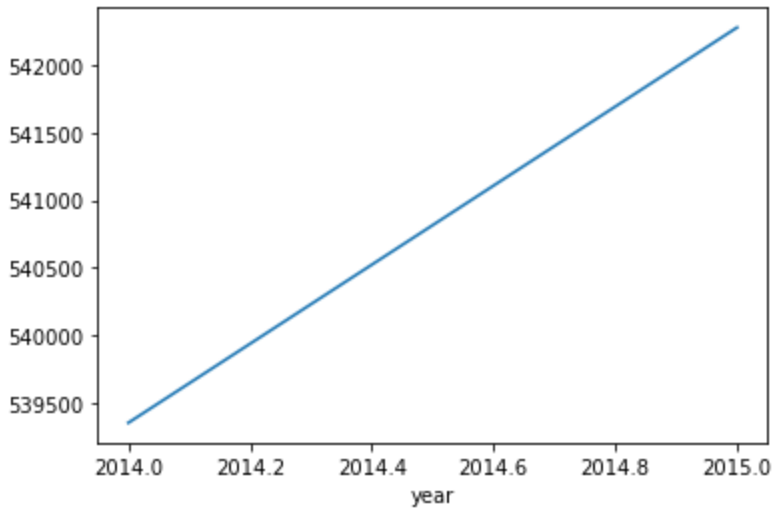
In [28]:
```
df.groupby('month').mean()['price'].plot()
```

Out[28]:
```
<AxesSubplot:xlabel='month'>
```



In [30]:
```
df.groupby('year').mean()['price'].plot()
```

Out[30]:
```
<AxesSubplot:xlabel='year'>
```



In [31]:
```
df =df.drop('date', axis =1)
```

In [32]:
```
df.columns
```

Out[32]:
```
Index(['price', 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors',
       'waterfront', 'view', 'condition', 'grade', 'sqft_above',
       'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode', 'lat', 'long',
       'sqft_living15', 'sqft_lot15', 'year', 'month'],
      dtype='object')
```

In [33]:
```
df.head()
```

Out[33]:

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	...	sqft_basem
0	221900.0	3	1.00	1180	5650	1.0	0	0	3	7	...	
1	538000.0	3	2.25	2570	7242	2.0	0	0	3	7	...	4
2	180000.0	2	1.00	770	10000	1.0	0	0	3	6	...	
3	604000.0	4	3.00	1960	5000	1.0	0	0	5	7	...	9
4	510000.0	3	2.00	1680	8080	1.0	0	0	3	8	...	

5 rows × 21 columns

In [36]:
```
df['zipcode'].value_counts()
```

Out[36]:
```
98103    602
98038    589
98115    583
98052    574
98117    553
        ...
98102    104
98010    100
98024     80
98148     57
98039     50
Name: zipcode, Length: 70, dtype: int64
```

In [38]:
```
# Nhìn vào sự ảnh hưởng của zipcode sẽ gây khó khăn cho mô hình học máy nên có thể bỏ qua
df = df.drop('zipcode',axis=1)
```

In [39]:
```
df['yr_renovated'].value_counts()
#Nhận thấy
```

Out[39]:
```
0       20683
2014       91
2013       37
2003       36
2005       35
          ...
1951        1
1959        1
1948        1
1954        1
1944        1
Name: yr_renovated, Length: 70, dtype: int64
```

In [40]:
```
df['sqft_basement'].value_counts()
```

Out[40]:
```
0      13110
600      221
700      218
500      214
800      206
        ...
518        1
374        1
784        1
906        1
248        1
Name: sqft_basement, Length: 306, dtype: int64
```