In [16]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [17]:
```
df=pd.read_csv("C:/Users/sirvi/OneDrive/Desktop/US HOUSING SALES MAIN FILE.csv")
#FILE UPLOAD
```

In [12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21613 entries, 0 to 21612
Data columns (total 30 columns):
 #   Column                      Non-Null Count  Dtype  
---  ------                      --------------  -----  
 0   count_no                    21613 non-null  int64  
 1   id                          21613 non-null  int64  
 2   date                        21613 non-null  object 
 3   price_RANGE                 21613 non-null  object 
 4   price                       21613 non-null  int64  
 5   BEDROOM_RANGE               21613 non-null  object 
 6   bedrooms                    21613 non-null  int64  
 7   bathrooms                   21603 non-null  float64
 8   sqft_living_AREA            21613 non-null  object 
 9   sqft_living                 21613 non-null  int64  
 10  sqft_lot_CATEGORICAL        21613 non-null  object 
 11  sqft_lot                    21613 non-null  int64  
 12  floors                      21613 non-null  float64
 13  waterfront                  21613 non-null  int64  
 14  view                        21613 non-null  int64  
 15  condition                   21613 non-null  int64  
 16  grade                       21613 non-null  int64  
 17  sqft_above_CATEGORICAL      21613 non-null  object 
 18  sqft_above                  21613 non-null  int64  
 19  sqft_basement_CATEGORICAL   21613 non-null  object 
 20  sqft_basement               21613 non-null  int64  
 21  yr_built_CATEGORICAL        21613 non-null  object 
 22  yr_built                    21613 non-null  int64  
 23  yr_renovated_CATEGORICAL    21613 non-null  object 
 24  yr_renovated                21613 non-null  int64  
 25  zipcode                     21613 non-null  int64  
 26  lat                         21613 non-null  float64
 27  long                        21613 non-null  float64
 28  sqft_living15               21613 non-null  int64  
 29  sqft_lot15                  21613 non-null  int64  
dtypes: float64(4), int64(17), object(9)
memory usage: 4.9+ MB
```

In [18]:
```
df.shape
```

Out[18]:
```
(21613, 30)
```

the data has 21613 rows and 30 columns

In [14]:
```
pd.isnull(df)
```

Out[14]:

	count_no	id	date	price_RANGE	price	BEDROOM_RANGE	bedrooms	bathrooms	sqft_living_AREA	sqft_living	...	sqft_baseme
0	False	False	False	False	False	False	False	False	False	False	...	Fa
1	False	False	False	False	False	False	False	False	False	False	...	Fa
2	False	False	False	False	False	False	False	False	False	False	...	Fa
3	False	False	False	False	False	False	False	False	False	False	...	Fa
4	False	False	False	False	False	False	False	False	False	False	...	Fa
...	...	...	...	...	...	...	...	...	...	...	...	
21608	False	False	False	False	False	False	False	False	False	False	...	Fa
21609	False	False	False	False	False	False	False	False	False	False	...	Fa
21610	False	False	False	False	False	False	False	False	False	False	...	Fa
21611	False	False	False	False	False	False	False	False	False	False	...	Fa
21612	False	False	False	False	False	False	False	False	False	False	...	Fa

21613 rows × 30 columns

no null data found

In [15]:
```
df.columns
```

Out[15]:
```
Index(['count_no', 'id', 'date', 'price_RANGE', 'price', 'BEDROOM_RANGE',
       'bedrooms', 'bathrooms', 'sqft_living_AREA', 'sqft_living',
       'sqft_lot_CATEGORICAL', 'sqft_lot', 'floors', 'waterfront', 'view',
       'condition', 'grade', 'sqft_above_CATEGORICAL', 'sqft_above',
       'sqft_basement_CATEGORICAL', 'sqft_basement', 'yr_built_CATEGORICAL',
       'yr_built', 'yr_renovated_CATEGORICAL', 'yr_renovated', 'zipcode',
       'lat', 'long', 'sqft_living15', 'sqft_lot15'],
      dtype='object')
```

DATA CLEANING ,BINNING,INTEGRATION COMPLETES HERE.

In [ ]:

EDA-EXPLORATORY DATA ANALYSIS

In [ ]:

In [19]:
```
df[['price','bedrooms']].describe()
```

Out[19]:

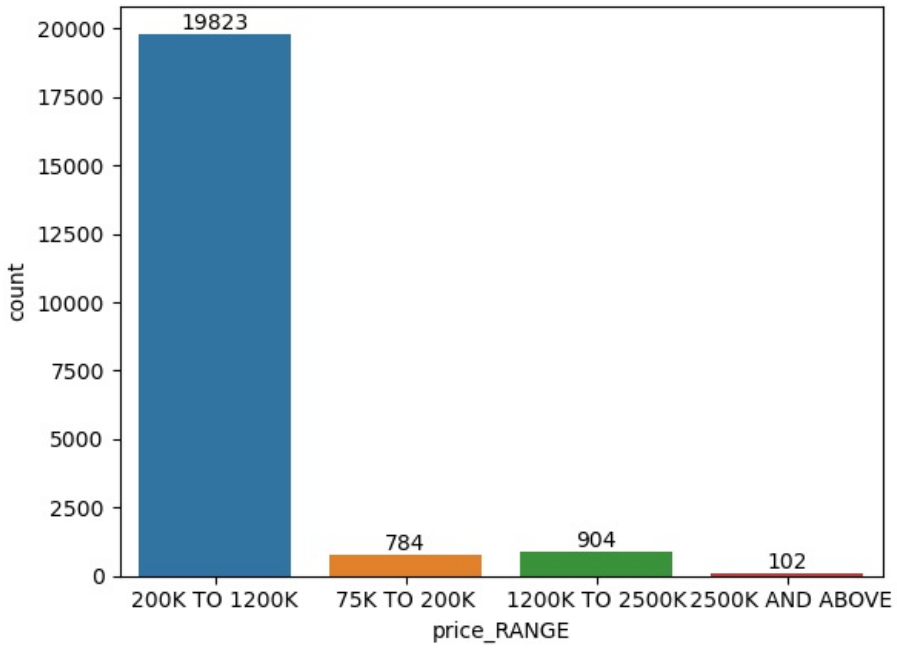
	price	bedrooms
count	2.161300e+04	21613.000000
mean	5.400881e+05	3.374404
std	3.671272e+05	0.929156
min	7.500000e+04	1.000000
25%	3.219500e+05	3.000000
50%	4.500000e+05	3.000000
75%	6.450000e+05	4.000000
max	7.700000e+06	33.000000

1. SALE AS PER PRICE RANGE--

200000 TO 1200000 USD HAS THE MOST NUMBER OF SALES ,WHILE 2500000 $ HAS THE LEAST NO OF SALES.

In [ ]:

In [20]:
```
bx=sns.countplot(x = 'price_RANGE',data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```

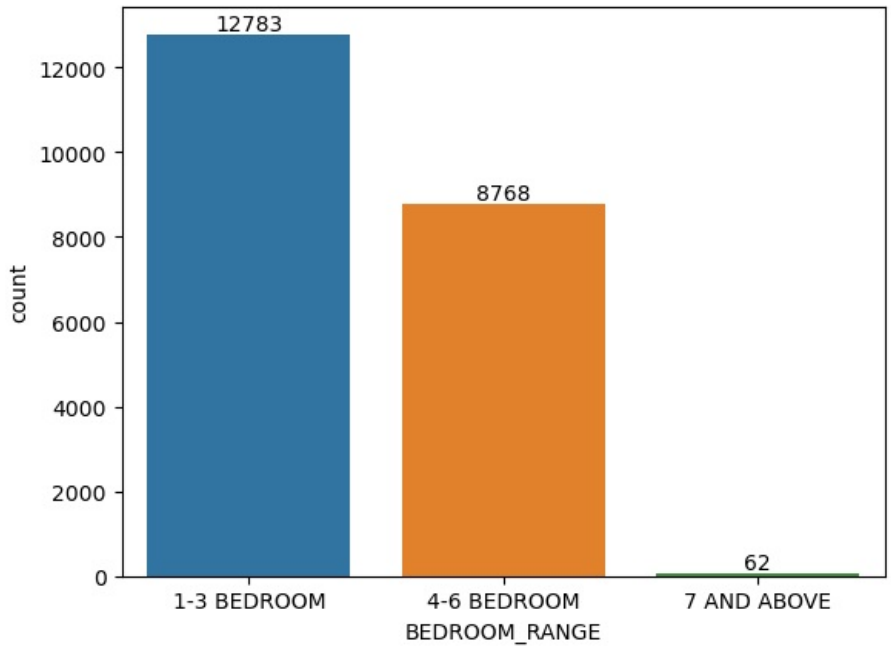


In [ ]:

2.SALE AS PER BEDROOMS-

MOST PEOPLE PREFER 1-3 BEDROOMS WHILE LUXURY ONES HAVING MORE THAN 7 BEDROOMS HAVE VERY FEW SALES i.e 62

In [22]:
```
bx=sns.countplot(x ='BEDROOM_RANGE' ,data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```



In [ ]:
```
```

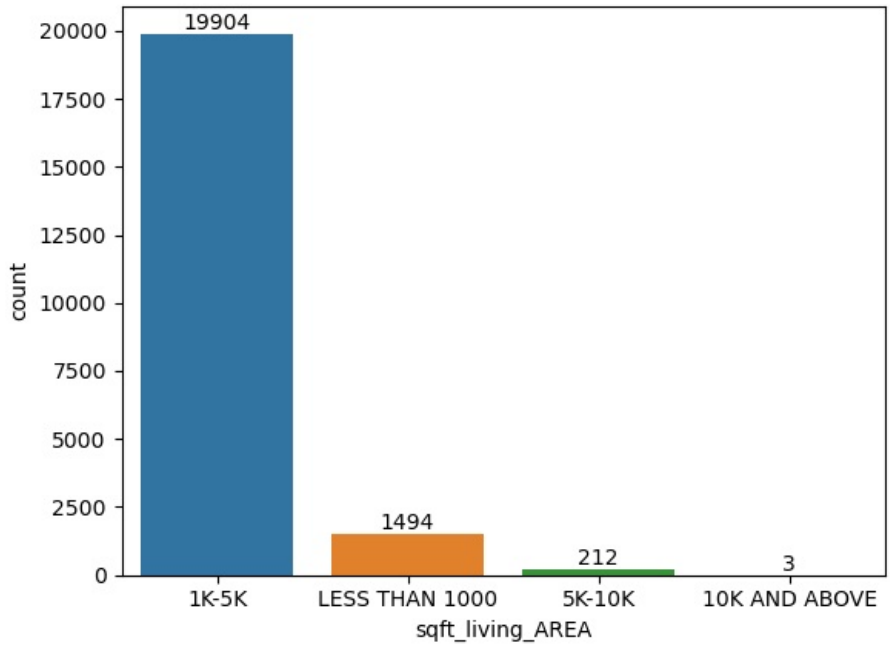
In [23]:
```
df.columns
```

Out[23]:
```
Index(['count_no', 'id', 'date', 'price_RANGE', 'price', 'BEDROOM_RANGE',
       'bedrooms', 'bathrooms', 'sqft_living_AREA', 'sqft_living',
       'sqft_lot_CATEGORICAL', 'sqft_lot', 'floors', 'waterfront', 'view',
       'condition', 'grade', 'sqft_above_CATEGORICAL', 'sqft_above',
       'sqft_basement_CATEGORICAL', 'sqft_basement', 'yr_built_CATEGORICAL',
       'yr_built', 'yr_renovated_CATEGORICAL', 'yr_renovated', 'zipcode',
       'lat', 'long', 'sqft_living15', 'sqft_lot15'],
      dtype='object')
```

In [ ]:
```
```

3.SALE AS PER SQ_FT LIVING AREA- PEOPLE MOSTLY PREFER 1K-5K SQUARE FEET LIVING AREA

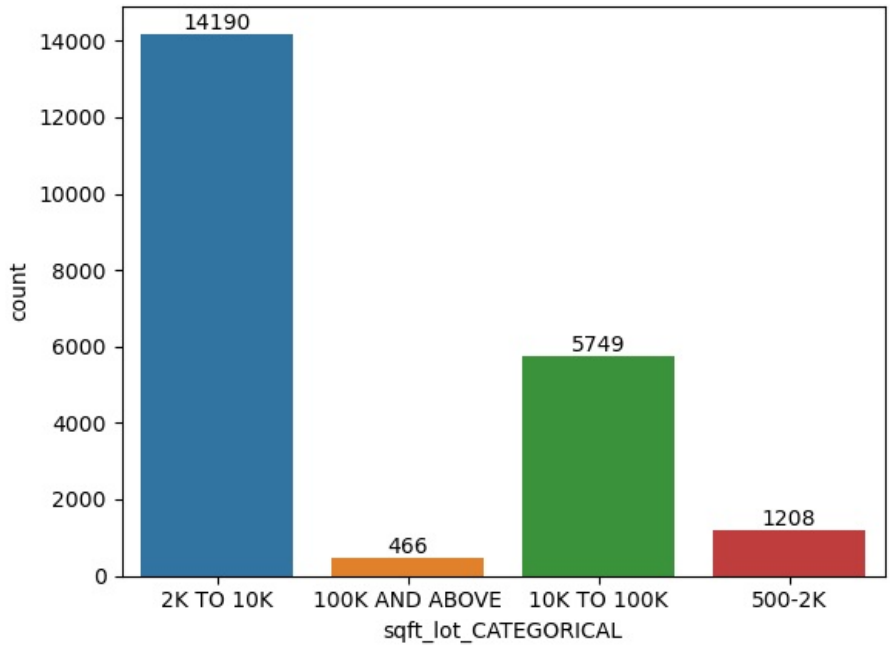
In [24]:
```
bx=sns.countplot(x = 'sqft_living_AREA',data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```



In [ ]:

4. SALE AS PER SQ_FEET PRICE- PEOPLE USUALLY PREFER TO BUY AT 2000 TO 10000 USD.

In [25]:
```
bx=sns.countplot(x ='sqft_lot_CATEGORICAL'
,data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```



In [ ]:

5.SALE AS PER CATEGORICAL- PEOPLE USED TO BUY THE MID_TERM CATEGORY FLATS MORE THAN THE HIGH CATEGORY FLATS.

In [ ]:

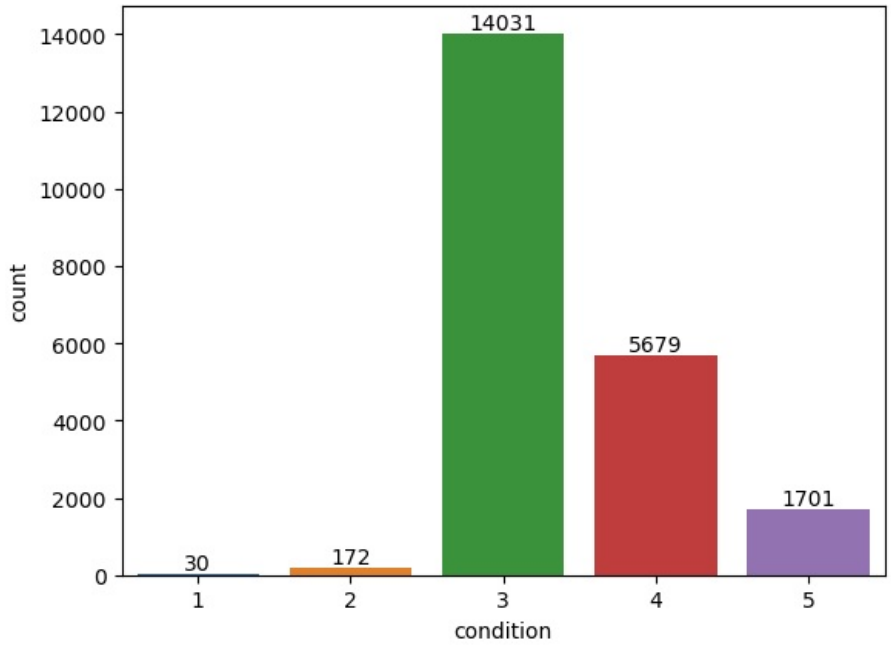
In [28]:
```
df.columns
```

Out[28]:
```
Index(['count_no', 'id', 'date', 'price_RANGE', 'price', 'BEDROOM_RANGE',
       'bedrooms', 'bathrooms', 'sqft_living_AREA', 'sqft_living',
       'sqft_lot_CATEGORICAL', 'sqft_lot', 'floors', 'waterfront', 'view',
       'condition', 'grade', 'sqft_above_CATEGORICAL', 'sqft_above',
       'sqft_basement_CATEGORICAL', 'sqft_basement', 'yr_built_CATEGORICAL',
       'yr_built', 'yr_renovated_CATEGORICAL', 'yr_renovated', 'zipcode',
       'lat', 'long', 'sqft_living15', 'sqft_lot15'],
      dtype='object')
```

In [ ]:

In [ ]:

In [29]:
```
bx=sns.countplot(x ='condition' ,data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```

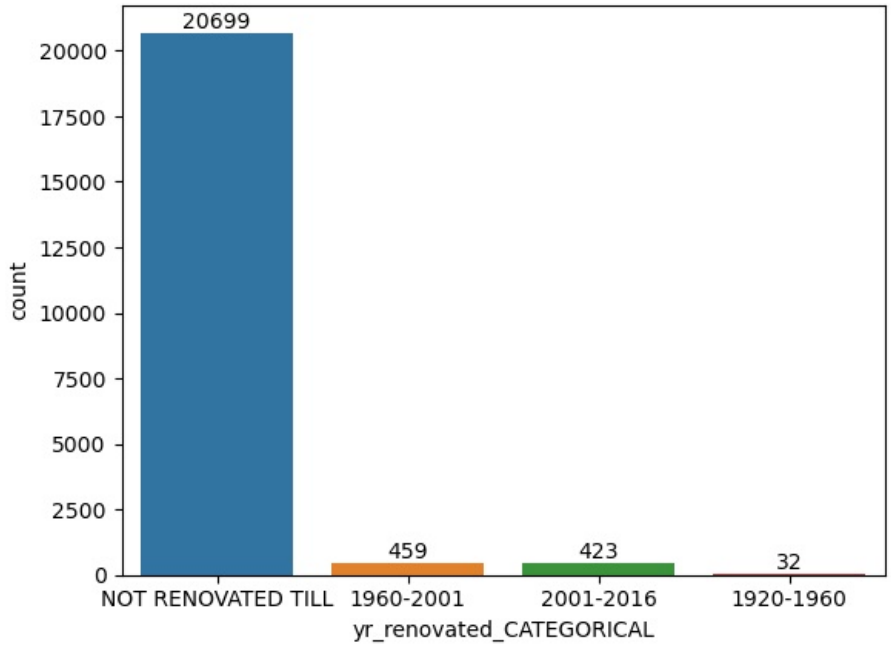


In [ ]:

6 PEOPLE LOVE TO BUY HOUSES WHICH IS IN ITS ORGANIC FORM

In [ ]:

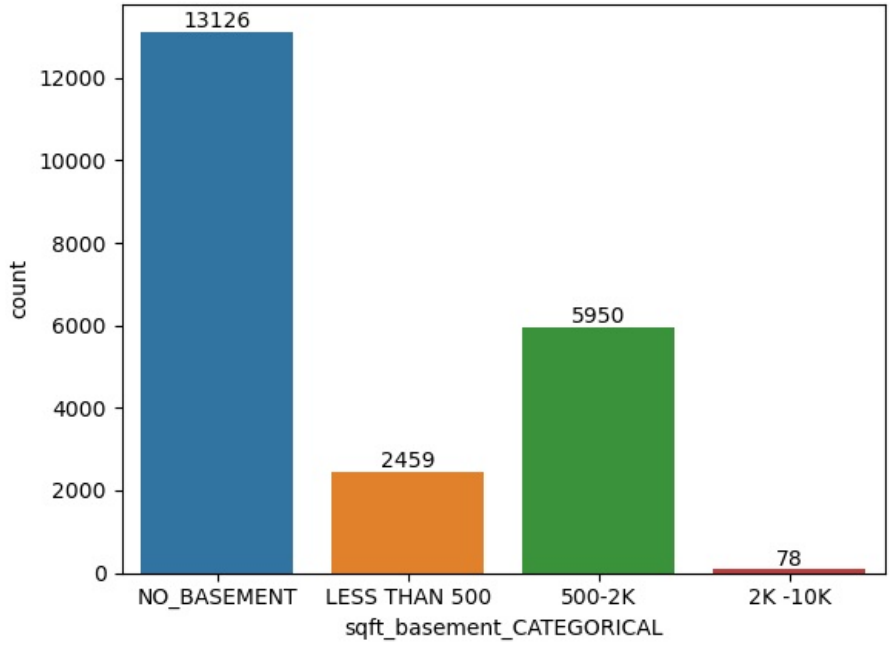
In [30]:
```
bx=sns.countplot(x ='yr_renovated_CATEGORICAL' ,data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```



In [ ]:

7. NO BASEMENT FLAT HAVE BETTER SALES REPRESENTATIVE AS THEY ARE CHEAPER AND AFFORDABLE

In [31]:
```
bx=sns.countplot(x = 'sqft_basement_CATEGORICAL',data=df)
for bars in bx.containers:
    bx.bar_label(bars)
```



In [ ]:

--------the end-----

In [ ]:

In [ ]:

In [ ]: