In [1]:

```python
import pandas as pd
```

In [3]:

```python
data=pd.read_csv("kc_house_data.csv")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_2664\2078774848.py in <module>
----> 1 data=pd.read_csv("kc_house_data.csv")

C:\ProgramData\Anaconda3\lib\site-packages\pandas\util\_decorators.py in wrapper(*args, **kwargs)
    309                     stacklevel=stacklevel,
    310                 )
--> 311             return func(*args, **kwargs)
    312 
    313         return wrapper

C:\ProgramData\Anaconda3\lib\site-packages\pandas\io\parsers\readers.py in read_csv(filepath_or_buffer, sep, delimiter, header, names, index_col, usecols, squeeze, prefix, mangle_dupe_cols, dtype, engine, converters, true_values, false_values, skipinitialspace, skiprows, skipfooter, nrows, na_values, keep_default_na, na_filter, verbose, skip_blank_lines, parse_dates, infer_datetime_format, keep_date_col, date_parser, dayfirst, cache_dates, iterator, chunksize, compression, thousands, decimal, lineterminator, quotechar, quoting, doublequote, escapechar, comment, encoding, encoding_errors, dialect, error_bad_lines, warn_bad_lines, on_bad_lines, delim_whitespace, low_memory, memory_map, float_precision, storage_options)
    676     kwds.update(kwds_defaults)
    677 
--> 678     return _read(filepath_or_buffer, kwds)
    679 
    680 

C:\ProgramData\Anaconda3\lib\site-packages\pandas\io\parsers\readers.py in _read(filepath_or_buffer, kwds)
    573 
    574     # Create the parser.
--> 575     parser = TextFileReader(filepath_or_buffer, **kwds)
    576 
    577     if chunksize or iterator:

C:\ProgramData\Anaconda3\lib\site-packages\pandas\io\parsers\readers.py in __init__(self, f, engine, **kwds)
    930 
    931         self.handles: IOHandles | None = None
--> 932         self._engine = self._make_engine(f, self.engine)
    933 
    934     def close(self):

C:\ProgramData\Anaconda3\lib\site-packages\pandas\io\parsers\readers.py in _make_engine(self, f, engine)
   1214             # "Union[str, PathLike[str], ReadCsvBuffer[bytes], ReadCsvBuffer[str]]"
   1215             # , "str", "bool", "Any", "Any", "Any", "Any", "Any"
-> 1216             self.handles = get_handle(  # type: ignore[call-overload]
   1217                 f,
   1218                 mode,

C:\ProgramData\Anaconda3\lib\site-packages\pandas\io\common.py in get_handle(path_or_buf, mode, encoding, compression, memory_map, is_text, errors, storage_options)
    784         if ioargs.encoding and "b" not in ioargs.mode:
    785             # Encoding
--> 786             handle = open(
    787                 handle,
    788                 ioargs.mode,

FileNotFoundError: [Errno 2] No such file or directory: 'kc_house_data.csv'
```

```
In [4]: import os
```

```
In [5]: os.getcwd()
```

```
Out[5]: 'C:\\Users\\dsund\\Untitled Folder 1'
```

```
In [7]: df=pd.read_csv("kc_house_data.csv")
```

```
In [8]: df
```

Out[8]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floor
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5650	1.(
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7242	2.(
2	5631500400	20150225T000000	180000.0	2	1.00	770	10000	1.(
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5000	1.(
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8080	1.(
...	...	...	...	...	...	...	...	..
21608	263000018	20140521T000000	360000.0	3	2.50	1530	1131	3.(
21609	6600060120	20150223T000000	400000.0	4	2.50	2310	5813	2.(
21610	1523300141	20140623T000000	402101.0	2	0.75	1020	1350	2.(
21611	291310100	20150116T000000	400000.0	3	2.50	1600	2388	2.(
21612	1523300157	20141015T000000	325000.0	2	0.75	1020	1076	2.(

21613 rows × 21 columns

```
In [9]: df.count()
```

```
Out[9]: id               21613
        date             21613
        price            21613
        bedrooms         21613
        bathrooms        21613
        sqft_living      21613
        sqft_lot         21613
        floors           21613
        waterfront       21613
        view             21613
        condition        21613
        grade            21613
        sqft_above       21613
        sqft_basement    21613
        yr_built         21613
        yr_renovated     21613
        zipcode          21613
        lat              21613
        long             21613
        sqft_living15    21613
        sqft_lot15       21613
        dtype: int64
```

```
In [11]: df.isnull().sum()
```

Out[11]:

```
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 [12]:

```
import seaborn as sns
```

In [13]:

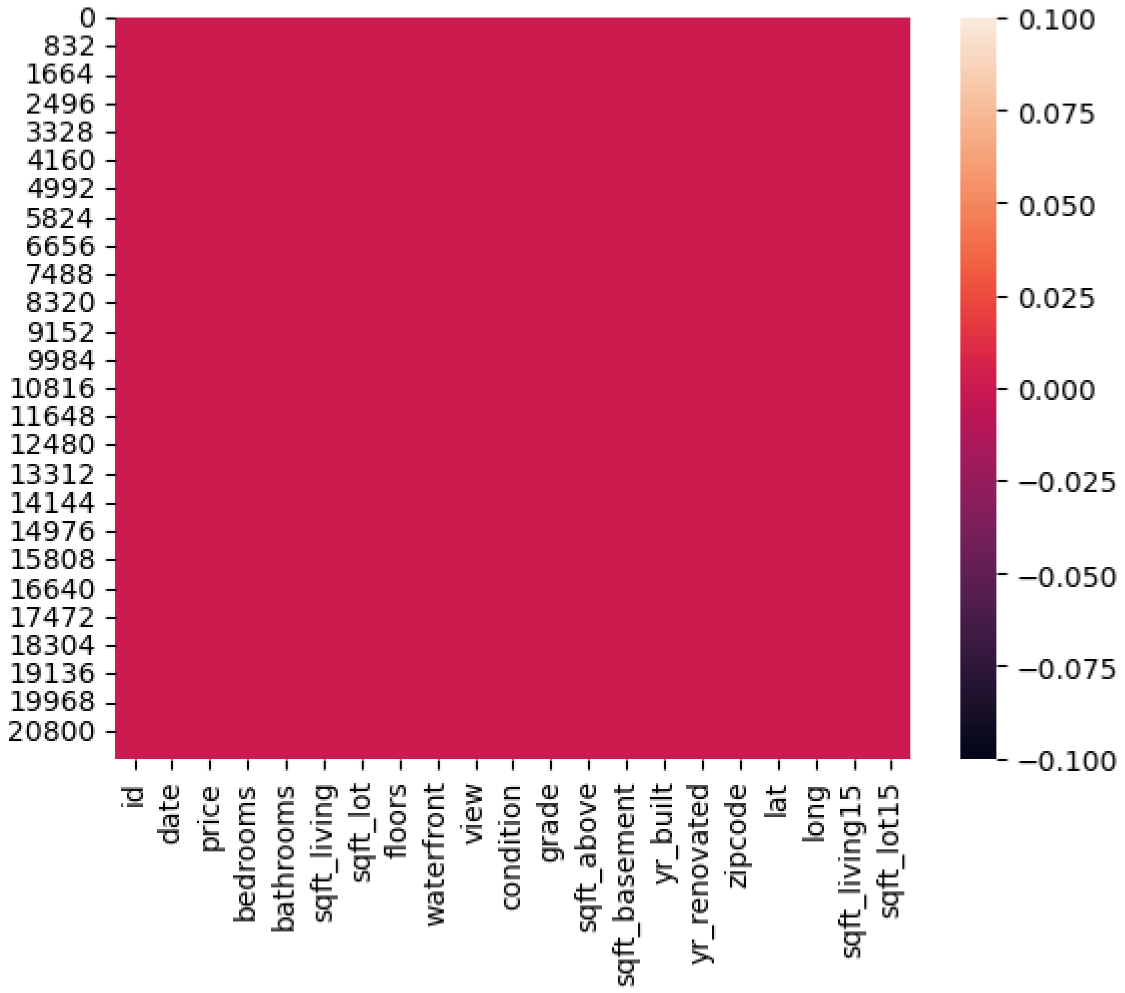
```
import matplotlib.pyplot as plt
```

In [14]:

```
sns.heatmap(df.isnull())
```

Out[14]:

```
<AxesSubplot:>
```



In [15]:
```
import numpy
```

In [16]:
```
import pandas
```

In [17]:
```
import csv
```

In [19]:
```
def _read_csv("kc_house_data"):
```

```
  File "C:\Users\dsund\AppData\Local\Temp\ipykernel_2664\519092208.py", line 1
    def _read_csv("kc_house_data"):
                  ^
SyntaxError: invalid syntax
```

In [22]:
```
df.head
```

Out[22]:

```
<bound method NDFrame.head of                id              date     price  bedroo
ms  bathrooms  \
0      7129300520  20141013T000000  221900.0         3       1.00
1      6414100192  20141209T000000  538000.0         3       2.25
2      5631500400  20150225T000000  180000.0         2       1.00
3      2487200875  20141209T000000  604000.0         4       3.00
4      1954400510  20150218T000000  510000.0         3       2.00
...           ...              ...       ...       ...        ...
21608   263000018  20140521T000000  360000.0         3       2.50
21609  6600060120  20150223T000000  400000.0         4       2.50
21610  1523300141  20140623T000000  402101.0         2       0.75
21611   291310100  20150116T000000  400000.0         3       2.50
21612  1523300157  20141015T000000  325000.0         2       0.75

       sqft_living  sqft_lot  floors  waterfront  view  ...  grade  \
0             1180      5650     1.0           0     0  ...      7
1             2570      7242     2.0           0     0  ...      7
2              770     10000     1.0           0     0  ...      6
3             1960      5000     1.0           0     0  ...      7
4             1680      8080     1.0           0     0  ...      8
...            ...       ...     ...         ...   ...  ...    ...
21608         1530      1131     3.0           0     0  ...      8
21609         2310      5813     2.0           0     0  ...      8
21610         1020      1350     2.0           0     0  ...      7
21611         1600      2388     2.0           0     0  ...      8
21612         1020      1076     2.0           0     0  ...      7

       sqft_above  sqft_basement  yr_built  yr_renovated  zipcode      lat  \
0            1180              0      1955             0    98178  47.5112
1            2170            400      1951          1991    98125  47.7210
2             770              0      1933             0    98028  47.7379
3            1050            910      1965             0    98136  47.5208
4            1680              0      1987             0    98074  47.6168
...           ...            ...       ...           ...      ...      ...
21608        1530              0      2009             0    98103  47.6993
21609        2310              0      2014             0    98146  47.5107
21610        1020              0      2009             0    98144  47.5944
21611        1600              0      2004             0    98027  47.5345
21612        1020              0      2008             0    98144  47.5941

          long  sqft_living15  sqft_lot15
0     -122.257           1340        5650
1     -122.319           1690        7639
2     -122.233           2720        8062
3     -122.393           1360        5000
4     -122.045           1800        7503
...        ...            ...         ...
21608 -122.346           1530        1509
21609 -122.362           1830        7200
21610 -122.299           1020        2007
21611 -122.069           1410        1287
21612 -122.299           1020        1357

[21613 rows x 21 columns]>
```

In [24]:

```
df.dtypes
```

Out[24]:
```
id                 int64
date              object
price            float64
bedrooms           int64
bathrooms        float64
sqft_living        int64
sqft_lot           int64
floors           float64
waterfront         int64
view               int64
condition          int64
grade              int64
sqft_above         int64
sqft_basement      int64
yr_built           int64
yr_renovated       int64
zipcode            int64
lat              float64
long             float64
sqft_living15      int64
sqft_lot15         int64
dtype: object
```

In [26]:
```
df['date']=df['date'].apply(pd.to_datetime)
```

In [27]:
```
df.dtypes
```

Out[27]:
```
id                        int64
date             datetime64[ns]
price                   float64
bedrooms                  int64
bathrooms               float64
sqft_living               int64
sqft_lot                  int64
floors                  float64
waterfront                int64
view                      int64
condition                 int64
grade                     int64
sqft_above                int64
sqft_basement             int64
yr_built                  int64
yr_renovated              int64
zipcode                   int64
lat                     float64
long                    float64
sqft_living15             int64
sqft_lot15                int64
dtype: object
```

In [28]:
```
df.head()
```

Out[28]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	v
0	7129300520	2014-10-13	221900.0	3	1.00	1180	5650	1.0	0	
1	6414100192	2014-12-09	538000.0	3	2.25	2570	7242	2.0	0	
2	5631500400	2015-02-25	180000.0	2	1.00	770	10000	1.0	0	
3	2487200875	2014-12-09	604000.0	4	3.00	1960	5000	1.0	0	
4	1954400510	2015-02-18	510000.0	3	2.00	1680	8080	1.0	0	

5 rows × 21 columns

In [29]:

```
df['year']=df.date.dt.year
```

In [30]:

```
df
```

Out[30]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfro
0	7129300520	2014-10-13	221900.0	3	1.00	1180	5650	1.0	
1	6414100192	2014-12-09	538000.0	3	2.25	2570	7242	2.0	
2	5631500400	2015-02-25	180000.0	2	1.00	770	10000	1.0	
3	2487200875	2014-12-09	604000.0	4	3.00	1960	5000	1.0	
4	1954400510	2015-02-18	510000.0	3	2.00	1680	8080	1.0	
...	...	...	...	...	...	...	...	...	
21608	263000018	2014-05-21	360000.0	3	2.50	1530	1131	3.0	
21609	6600060120	2015-02-23	400000.0	4	2.50	2310	5813	2.0	
21610	1523300141	2014-06-23	402101.0	2	0.75	1020	1350	2.0	
21611	291310100	2015-01-16	400000.0	3	2.50	1600	2388	2.0	
21612	1523300157	2014-10-15	325000.0	2	0.75	1020	1076	2.0	

21613 rows × 22 columns

In [32]:

```
df.head()
```

Out[32]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	v
0	7129300520	2014-10-13	221900.0	3	1.00	1180	5650	1.0	0	
1	6414100192	2014-12-09	538000.0	3	2.25	2570	7242	2.0	0	
2	5631500400	2015-02-25	180000.0	2	1.00	770	10000	1.0	0	
3	2487200875	2014-12-09	604000.0	4	3.00	1960	5000	1.0	0	
4	1954400510	2015-02-18	510000.0	3	2.00	1680	8080	1.0	0	

5 rows × 22 columns

In [33]:
```
df.groupby('year').price.max()
```

Out[33]:
```
year
2014    7700000.0
2015    5350000.0
Name: price, dtype: float64
```

In [34]:
```
df.groupby('year').price.min()
```

Out[34]:
```
year
2014    78000.0
2015    75000.0
Name: price, dtype: float64
```

In [36]:
```
df.groupby('year').price.mean()
```

Out[36]:
```
year
2014    539181.428415
2015    541988.992264
Name: price, dtype: float64
```

In [ ]: