In [1]:
```
# London Housing Dataset
# . Monthly average house price
# . Yearly number of houses sold
# . Monthly number of crimes committed
```

In [29]:
```
import pandas as pd
```

In [30]:
```
data = pd.read_csv('London Housing Data.csv')
```

In [31]:
```
data
```

Out[31]:

| | date | area | average_price | code | houses_sold | no_of_crimes |
|---|---|---|---|---|---|---|
| **0** | 1/1/1995 | city of london | 91449 | E09000001 | 17.0 | NaN |
| **1** | 2/1/1995 | city of london | 82203 | E09000001 | 7.0 | NaN |
| **2** | 3/1/1995 | city of london | 79121 | E09000001 | 14.0 | NaN |
| **3** | 4/1/1995 | city of london | 77101 | E09000001 | 7.0 | NaN |
| **4** | 5/1/1995 | city of london | 84409 | E09000001 | 10.0 | NaN |
| **...** | ... | ... | ... | ... | ... | ... |
| **13544** | 9/1/2019 | england | 249942 | E92000001 | 64605.0 | NaN |
| **13545** | 10/1/2019 | england | 249376 | E92000001 | 68677.0 | NaN |
| **13546** | 11/1/2019 | england | 248515 | E92000001 | 67814.0 | NaN |
| **13547** | 12/1/2019 | england | 250410 | E92000001 | NaN | NaN |
| **13548** | 1/1/2020 | england | 247355 | E92000001 | NaN | NaN |

13549 rows × 6 columns

In [32]:
```
data.count()
```

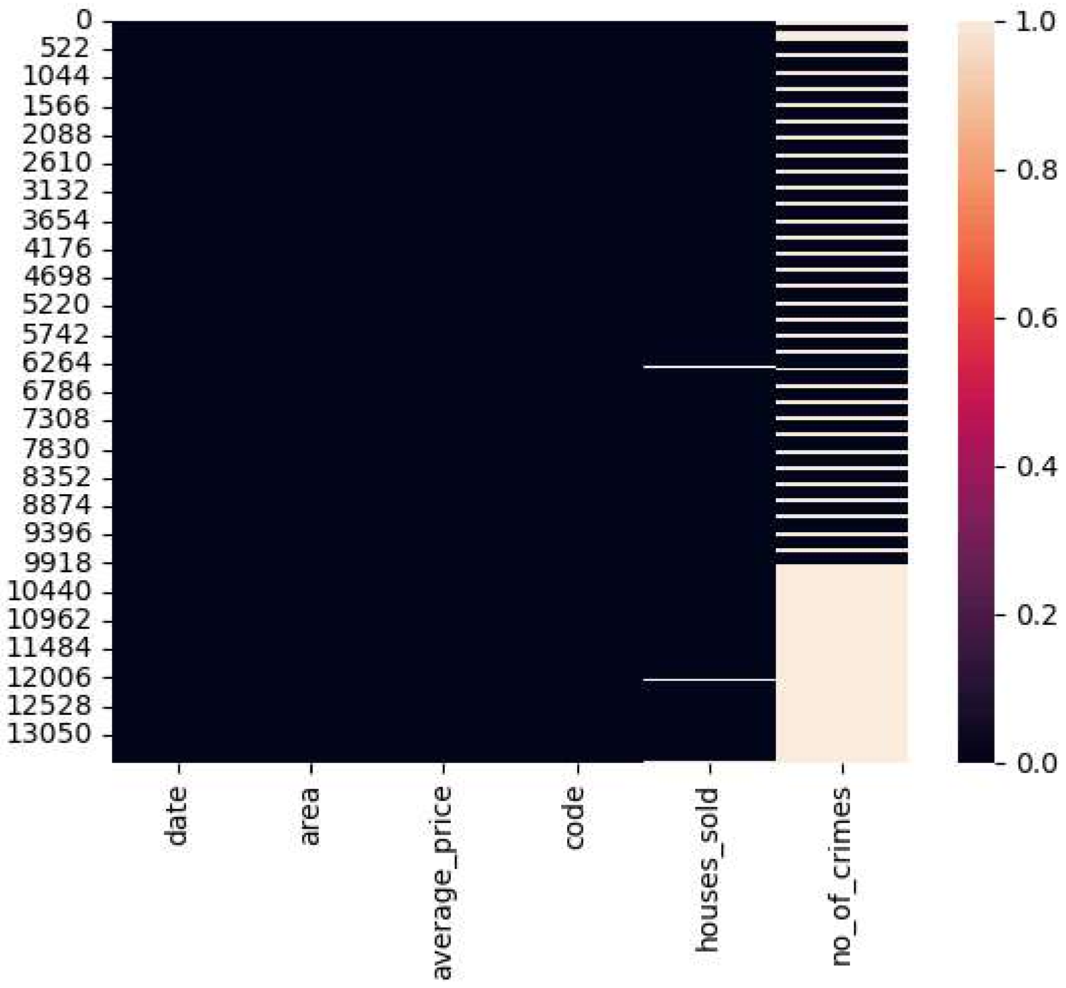
Out[32]:
```
date             13549
area             13549
average_price    13549
code             13549
houses_sold      13455
no_of_crimes      7439
dtype: int64
```

In [33]:
```
data.isnull().sum()
```

Out[33]:
```
date                0
area                0
average_price       0
code                0
houses_sold        94
no_of_crimes     6110
dtype: int64
```

In [34]:
```
# to see null values in the form of heatmap

import seaborn as sns
import matplotlib.pyplot as plt
sns.heatmap(data.isnull())
plt.show()
```



## 1. Convert the datatype of 'Date' column to Date-Time format

In [35]:
```
data.dtypes
```

Out[35]:
```
date              object
area              object
average_price      int64
code              object
houses_sold      float64
no_of_crimes     float64
dtype: object
```

In [36]:
```
#data.date = pd.to_datetime(data.date)
```

In [37]:
```
data.date = pd.to_datetime(data.date)
```

In [38]:
```
data.dtypes
```

Out[38]:
```
date              datetime64[ns]
area                      object
average_price              int64
code                      object
houses_sold              float64
no_of_crimes             float64
dtype: object
```

## 2. Add a new column "year" in the dataframe, which contains years only.

In [39]:
```
# data['New_Column'] = df.Date_Column.dt.year   ** dt.year extracts the year f
```

In [40]:
```
data['year'] = data.date.dt.year
#data['month'] = data.date.dt.month
```

In [41]:
```
data
```

Out[41]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year |
|---|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 |
| **1** | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 |
| **2** | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 |
| **3** | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 |
| **4** | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **13544** | 2019-09-01 | england | 249942 | E92000001 | 64605.0 | NaN | 2019 |
| **13545** | 2019-10-01 | england | 249376 | E92000001 | 68677.0 | NaN | 2019 |
| **13546** | 2019-11-01 | england | 248515 | E92000001 | 67814.0 | NaN | 2019 |
| **13547** | 2019-12-01 | england | 250410 | E92000001 | NaN | NaN | 2019 |
| **13548** | 2020-01-01 | england | 247355 | E92000001 | NaN | NaN | 2020 |

13549 rows × 7 columns

## 2.1 Add a new column 'month' as 2nd column in the dataframe, which contains month only

In [42]:
```
# df.insert(index, 'new_column_name','new_column_values')
# data.insert (1, 'month', data.date.dt.month)
```

In [43]:
```
data.insert(1, 'month', data.date.dt.month)
```

In [44]:
```
data.head()
```

Out[44]:

| | date | month | area | average_price | code | houses_sold | no_of_crimes | year |
|---|---|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | 1 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 |
| **1** | 1995-02-01 | 2 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 |
| **2** | 1995-03-01 | 3 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 |
| **3** | 1995-04-01 | 4 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 |
| **4** | 1995-05-01 | 5 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 |

## 3. Remove the columns 'year' and 'month' from the dataframe

In [46]:
```
#data.drop(['month','year'], axis=1, inplace = True)
```

In [47]:
```
data.drop( ['month','year'], axis=1, inplace = True)
```

In [48]:
```
data.head()
```

Out[48]:

| | date | area | average_price | code | houses_sold | no_of_crimes |
|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN |
| **1** | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN |
| **2** | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN |
| **3** | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN |
| **4** | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN |

## 4. Show all the records where 'No. of Crimes' is 0.And , how many such records are there?

In [53]:
```
data[data.no_of_crimes == 0]
```

Out[53]:

| | date | area | average_price | code | houses_sold | no_of_crimes |
|---|---|---|---|---|---|---|
| **72** | 2001-01-01 | city of london | 284262 | E09000001 | 24.0 | 0.0 |
| **73** | 2001-02-01 | city of london | 198137 | E09000001 | 37.0 | 0.0 |
| **74** | 2001-03-01 | city of london | 189033 | E09000001 | 44.0 | 0.0 |
| **75** | 2001-04-01 | city of london | 205494 | E09000001 | 38.0 | 0.0 |
| **76** | 2001-05-01 | city of london | 223459 | E09000001 | 30.0 | 0.0 |
| **...** | ... | ... | ... | ... | ... | ... |
| **178** | 2009-11-01 | city of london | 397909 | E09000001 | 11.0 | 0.0 |
| **179** | 2009-12-01 | city of london | 411955 | E09000001 | 16.0 | 0.0 |
| **180** | 2010-01-01 | city of london | 464436 | E09000001 | 20.0 | 0.0 |
| **181** | 2010-02-01 | city of london | 490525 | E09000001 | 9.0 | 0.0 |
| **182** | 2010-03-01 | city of london | 498241 | E09000001 | 15.0 | 0.0 |

104 rows × 6 columns

In [54]:
```
len(data[data.no_of_crimes == 0])
```

Out[54]:
```
104
```

## 5. What is the maximum and minimum 'average_price' per year in england?

In [55]:
```
data.head(2)
```

Out[55]:

| | date | area | average_price | code | houses_sold | no_of_crimes |
|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN |
| **1** | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN |

In [59]:
```
data['year'] = data.date.dt.year
```

In [60]:
```
data.head()
```

Out[60]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year |
|---|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 |
| **1** | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 |
| **2** | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 |
| **3** | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 |
| **4** | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 |

In [64]:
```
df1 = data[data.area == 'england']
```

In [65]:
```
df1
```

Out[65]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year |
|---|---|---|---|---|---|---|---|
| **13248** | 1995-01-01 | england | 53203 | E92000001 | 47639.0 | NaN | 1995 |
| **13249** | 1995-02-01 | england | 53096 | E92000001 | 47880.0 | NaN | 1995 |
| **13250** | 1995-03-01 | england | 53201 | E92000001 | 67025.0 | NaN | 1995 |
| **13251** | 1995-04-01 | england | 53591 | E92000001 | 56925.0 | NaN | 1995 |
| **13252** | 1995-05-01 | england | 53678 | E92000001 | 64192.0 | NaN | 1995 |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **13544** | 2019-09-01 | england | 249942 | E92000001 | 64605.0 | NaN | 2019 |
| **13545** | 2019-10-01 | england | 249376 | E92000001 | 68677.0 | NaN | 2019 |
| **13546** | 2019-11-01 | england | 248515 | E92000001 | 67814.0 | NaN | 2019 |
| **13547** | 2019-12-01 | england | 250410 | E92000001 | NaN | NaN | 2019 |
| **13548** | 2020-01-01 | england | 247355 | E92000001 | NaN | NaN | 2020 |

301 rows × 7 columns

In [66]:
```
#df1 = data[data.area == 'england']
#df1.groupby('year').average_price.max()/min()/mean()
```

In [67]:
```
df1.groupby('year').average_price.max()
```

Out[67]:
```
year
1995     53901
1996     55755
1997     61564
1998     65743
1999     75071
2000     84191
2001     95992
2002    119982
2003    138985
2004    160330
2005    167244
2006    182031
2007    194764
2008    191750
2009    174136
2010    180807
2011    177335
2012    180129
2013    188544
2014    203639
2015    219582
2016    231922
2017    242628
2018    248620
2019    250410
2020    247355
Name: average_price, dtype: int64
```

In [68]:
```
df1.groupby('year').average_price.min()
```

Out[68]:
```
year
1995     52788
1996     52333
1997     55789
1998     61659
1999     65522
2000     75219
2001     84245
2002     96215
2003    121610
2004    139719
2005    158572
2006    166544
2007    181824
2008    165795
2009    159340
2010    174458
2011    173046
2012    174161
2013    176816
2014    188265
2015    202856
2016    220361
2017    231593
2018    240428
2019    243281
2020    247355
Name: average_price, dtype: int64
```

## 6. What is the Maximum and Minimum No. of Crimes recorded per area ?

In [69]:
```
data.area
```

Out[69]:
```
0        city of london
1        city of london
2        city of london
3        city of london
4        city of london
              ...
13544           england
13545           england
13546           england
13547           england
13548           england
Name: area, Length: 13549, dtype: object
```

```
In [70]: # data.groupby('area').no_of_crimes.max()
         # data.groupby('area').no_of_crimes.min().sort_values(ascending =True)
```

```
In [71]: data.groupby('area').no_of_crimes.max()
```

```
Out[71]: area
         barking and dagenham      2049.0
         barnet                    2893.0
         bexley                    1914.0
         brent                     2937.0
         bromley                   2637.0
         camden                    4558.0
         city of london              10.0
         croydon                   3263.0
         ealing                    3401.0
         east midlands                NaN
         east of england              NaN
         enfield                   2798.0
         england                      NaN
         greenwich                 2853.0
         hackney                   3466.0
         hammersmith and fulham    2645.0
         haringey                  3199.0
         harrow                    1763.0
         havering                  1956.0
         hillingdon                2819.0
         hounslow                  2817.0
         inner london                 NaN
         islington                 3384.0
         kensington and chelsea    2778.0
         kingston upon thames      1379.0
         lambeth                   4701.0
         lewisham                  2813.0
         london                       NaN
         merton                    1623.0
         newham                    3668.0
         north east                   NaN
         north west                   NaN
         outer london                 NaN
         redbridge                 2560.0
         richmond upon thames      1551.0
         south east                   NaN
         south west                   NaN
         southwark                 3821.0
         sutton                    1425.0
         tower hamlets             3316.0
         waltham forest            2941.0
         wandsworth                3051.0
         west midlands                NaN
         westminster               7461.0
         yorks and the humber         NaN
         Name: no_of_crimes, dtype: float64
```

In [72]:
```
data.groupby('area').no_of_crimes.min().sort_values(ascending = True)
```

Out[72]:
```
area
city of london                0.0
kingston upon thames        692.0
richmond upon thames        700.0
sutton                      787.0
merton                      819.0
bexley                      860.0
harrow                      937.0
havering                   1130.0
barking and dagenham       1217.0
hammersmith and fulham     1323.0
kensington and chelsea     1347.0
bromley                    1441.0
hillingdon                 1445.0
redbridge                  1487.0
greenwich                  1513.0
hounslow                   1529.0
haringey                   1536.0
waltham forest             1575.0
wandsworth                 1582.0
enfield                    1635.0
tower hamlets              1646.0
lewisham                   1675.0
barnet                     1703.0
brent                      1850.0
hackney                    1870.0
ealing                     1871.0
islington                  1871.0
croydon                    2031.0
camden                     2079.0
newham                     2130.0
southwark                  2267.0
lambeth                    2381.0
westminster                3504.0
east midlands                 NaN
east of england               NaN
england                       NaN
inner london                  NaN
london                        NaN
north east                    NaN
north west                    NaN
outer london                  NaN
south east                    NaN
south west                    NaN
west midlands                 NaN
yorks and the humber          NaN
Name: no_of_crimes, dtype: float64
```

## 7. Show the total count of records of each area, where average price is less than 100000.

In [73]:
```
#df1 = data[data.area == 'england']
#df1.groupby('year').average_price.max()/min()/mean()
```

In [74]:
```
data[data.average_price < 100000]
```

Out[74]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year |
|---|---|---|---|---|---|---|---|
| **0** | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 |
| **1** | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 |
| **2** | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 |
| **3** | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 |
| **4** | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **13330** | 2001-11-01 | england | 95083 | E92000001 | 109149.0 | NaN | 2001 |
| **13331** | 2001-12-01 | england | 95992 | E92000001 | 93329.0 | NaN | 2001 |
| **13332** | 2002-01-01 | england | 96215 | E92000001 | 71678.0 | NaN | 2002 |
| **13333** | 2002-02-01 | england | 96676 | E92000001 | 77131.0 | NaN | 2002 |
| **13334** | 2002-03-01 | england | 98962 | E92000001 | 102828.0 | NaN | 2002 |

2209 rows × 7 columns

In [ ]: