London Housing Dataset

This dataset is primarily centered around the housing market of London. It contains a lot of information- -Monthly Average House Price -Yearly Number of Houses Sold -Monthly Number of Crimes Committed

The dataset used here is from year 1995 to 2021 of each different area

The data is available in csv file

```
# importing pandas
import pandas as pd

#impritng csv file to notebbok using pandas
house=pd.read_csv('file.csv')

#checking the file
house

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

       no_of_crimes
0               NaN
1               NaN
2               NaN
```

```
3                NaN
4                NaN
...              ...
13544            NaN
13545            NaN
13546            NaN
13547            NaN
13548            NaN

[13549 rows x 6 columns]
```

```python
house.count()  #used to count all non null values in a column
```

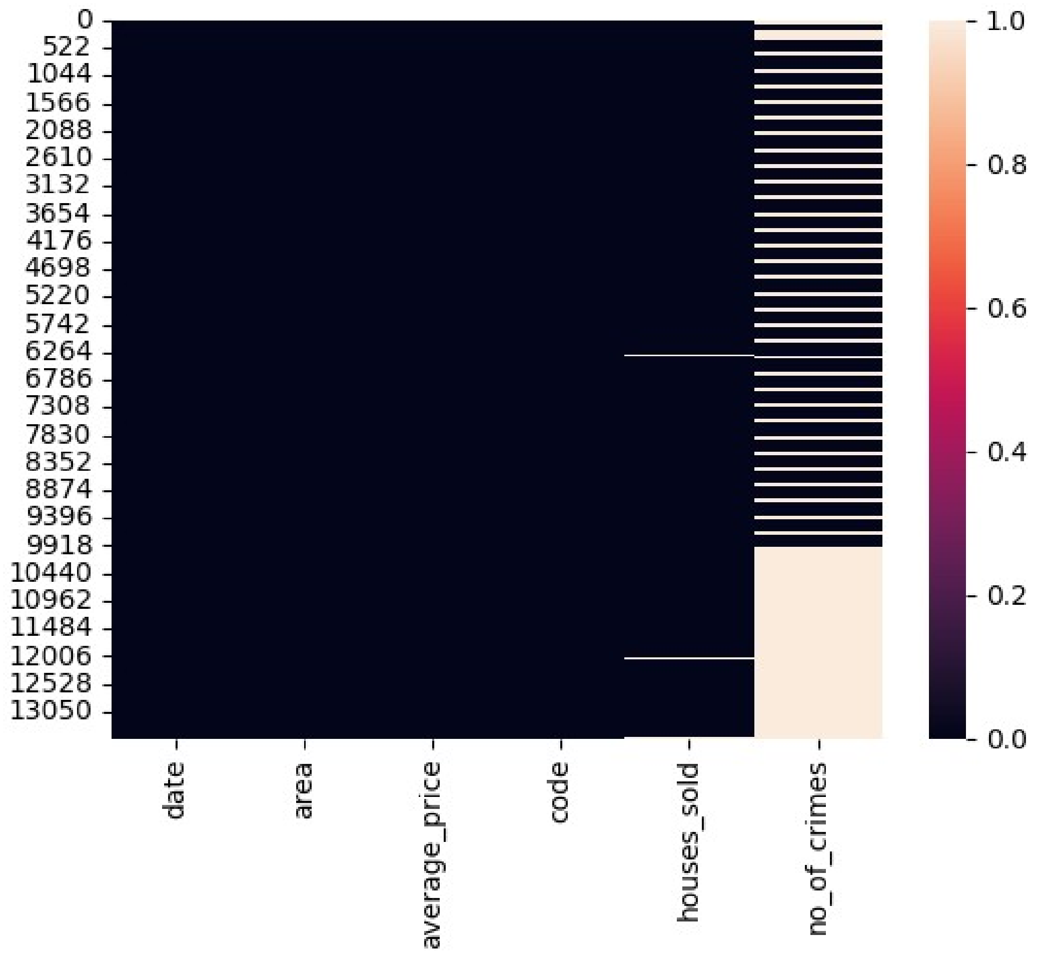
```
date             13549
area             13549
average_price    13549
code             13549
houses_sold      13455
no_of_crimes      7439
dtype: int64
```

```python
house.isnull().sum()  #used to count all null values in a column
```

```
date                0
area                0
average_price       0
code                0
houses_sold        94
no_of_crimes     6110
dtype: int64
```

```python
#for visualising this part we'll use seaborn
import seaborn as sns
import matplotlib.pyplot as plt

sns.heatmap(house.isnull())
plt.show()
```



White color is showing null values

(A) Convert the Datatype of 'Date' to Date-Time format.

```
#checking the datatypes
house.dtypes

date                  object
area                  object
average_price          int64
code                  object
houses_sold          float64
no_of_crimes         float64
dtype: object

house['date']=pd.to_datetime(house['date'])  #conerting the date ->
datetime

house.dtypes
```

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

(B1) Add a new column 'year' in dataframe which contain years only.

```
house

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

       no_of_crimes
0               NaN
1               NaN
2               NaN
3               NaN
4               NaN
...             ...
13544           NaN
13545           NaN
13546           NaN
13547           NaN
```

```
13548          NaN

[13549 rows x 6 columns]

house['year']=house['date'].dt.year   #extracting and adding a year
col to dataframe

house.head(2)

        date             area  average_price        code  houses_sold  \
0 1995-01-01  city of london          91449   E09000001         17.0
1 1995-02-01  city of london          82203   E09000001          7.0

   no_of_crimes  year
0           NaN  1995
1           NaN  1995
```

By default column added goes at the end but if you want to insert at certian location then do this let's move to another question

(B2) Add a new column 'Month' as 2nd column in dataframe, which contains month only.

```
#to perform this we''ll use inseet function
house.head(2)

        date             area  average_price        code  houses_sold  \
0 1995-01-01  city of london          91449   E09000001         17.0
1 1995-02-01  city of london          82203   E09000001          7.0

   no_of_crimes  year
0           NaN  1995
1           NaN  1995

house.insert(2,'month',house['date'].dt.month)
#df.insert(index,'new_col_name',new_col_values)

house.head(2)

        date             area  month  average_price        code
houses_sold  \
0 1995-01-01  city of london      1          91449   E09000001
17.0
1 1995-02-01  city of london      2          82203   E09000001
7.0

   no_of_crimes  year
0           NaN  1995
1           NaN  1995
```

(B3) Remove 'month' and 'year' column from the dataframe.

```
house.head(2)

         date             area  month  average_price       code
houses_sold  \
0 1995-01-01  city of london      1          91449  E09000001
17.0
1 1995-02-01  city of london      2          82203  E09000001
7.0

   no_of_crimes  year
0           NaN  1995
1           NaN  1995

house.drop(['month','year'],axis=1,inplace=True)

house.head(2)

         date             area  average_price       code  houses_sold  \
0 1995-01-01  city of london          91449  E09000001         17.0
1 1995-02-01  city of london          82203  E09000001          7.0

   no_of_crimes
0           NaN
1           NaN
```

(C) Show all the records where crime is 0.

```
house.head(5)

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

   no_of_crimes
0           NaN
1           NaN
2           NaN
3           NaN
4           NaN

house['no_of_crimes']==0

0        False
1        False
```

```
2        False
3        False
4        False
         ...
13544    False
13545    False
13546    False
13547    False
13548    False
Name: no_of_crimes, Length: 13549, dtype: bool
```

```
house[house['no_of_crimes']==0]
```

```
          date            area  average_price       code  houses_sold  \
72  2001-01-01  city of london         284262  E09000001         24.0

73  2001-02-01  city of london         198137  E09000001         37.0

74  2001-03-01  city of london         189033  E09000001         44.0

75  2001-04-01  city of london         205494  E09000001         38.0

76  2001-05-01  city of london         223459  E09000001         30.0

..         ...             ...            ...        ...          ...

178 2009-11-01  city of london         397909  E09000001         11.0

179 2009-12-01  city of london         411955  E09000001         16.0

180 2010-01-01  city of london         464436  E09000001         20.0

181 2010-02-01  city of london         490525  E09000001          9.0

182 2010-03-01  city of london         498241  E09000001         15.0


     no_of_crimes
72            0.0
73            0.0
74            0.0
75            0.0
76            0.0
..            ...
178           0.0
179           0.0
180           0.0
181           0.0
182           0.0
```

```
[104 rows x 6 columns]
```

```
len(house[house['no_of_crimes']==0])
```

```
104
```

(D) What is the max and min 'average_price' per year in England?

```
house.head(5)
```

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

   no_of_crimes  
0           NaN  
1           NaN  
2           NaN  
3           NaN  
4           NaN  
```

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

```
house1=house[house['area']=='england']
```

```
house1
```

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

       no_of_crimes  year  
13248           NaN  1995  
13249           NaN  1995  
13250           NaN  1995  
13251           NaN  1995  
```

```
13252           NaN  1995
...             ...   ...
13544           NaN  2019
13545           NaN  2019
13546           NaN  2019
13547           NaN  2019
13548           NaN  2020

[301 rows x 7 columns]
```

```
house1.groupby('year').average_price.max()
```

```
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
```

```
house1.groupby('year').average_price.min()
```

```
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
```

```
house1.groupby('year').average_price.mean()
```

```
year
1995     53322.416667
1996     54151.500000
1997     59160.666667
1998     64301.666667
1999     70070.750000
2000     80814.333333
2001     90306.750000
2002    107981.500000
2003    130218.583333
2004    152314.416667
2005    163570.000000
2006    174351.500000
2007    190025.583333
2008    182379.916667
2009    166558.666667
2010    177472.666667
2011    175230.000000
2012    177488.000000
2013    182581.416667
2014    197771.083333
2015    211174.750000
2016    227337.166667
2017    238161.166667
2018    245018.333333
2019    247101.083333
```

```
2020    247355.000000
Name: average_price, dtype: float64
```

(E) What is the max and min crime per year?

```
house.head(5)
```

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

   no_of_crimes  year
0           NaN  1995
1           NaN  1995
2           NaN  1995
3           NaN  1995
4           NaN  1995
```

```
house.groupby('area').no_of_crimes.max()
```

```
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
```

```
house.groupby('area').no_of_crimes.min().sort_values(ascending=True)
```

```
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
```

(F) Show the total count of records of each area, where avg price is 100000.

```
house[house['average_price']<100000].area.value_counts()
```

```
north east                  112
north west                  111
yorks and the humber        110
east midlands                96
west midlands                94
england                      87
barking and dagenham         85
south west                   78
east of england              76
newham                       72
bexley                       64
waltham forest               64
lewisham                     62
havering                     60
south east                   59
greenwich                    59
croydon                      57
enfield                      54
sutton                       54
hackney                      53
redbridge                    52
southwark                    48
tower hamlets                47
```

```
outer london             46
hillingdon               44
lambeth                  41
hounslow                 41
brent                    40
london                   39
merton                   35
haringey                 33
bromley                  33
inner london             31
ealing                   31
kingston upon thames     30
harrow                   30
wandsworth               26
barnet                   25
islington                19
city of london           11
Name: area, dtype: int64
```