```
import pandas as pd
```

```
data = pd.read_csv('London Housing Data.csv')
```

```
data
```

	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

```
data.head(5)
```

	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

```
data.columns
```

```
Index(['date', 'area', 'average_price', 'code', 'houses_sold', 'no_of_crimes'], dtype='object')
```

```
data.shape
```

```
(13549, 6)
```

```
data.count()
```

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

```
data.isnull()
```

	date	area	average_price	code	houses_sold	no_of_crimes
0	False	False	False	False	False	True
1	False	False	False	False	False	True
2	False	False	False	False	False	True
3	False	False	False	False	False	True
4	False	False	False	False	False	True
...	...	...	...	...	...	...
13544	False	False	False	False	False	True
13545	False	False	False	False	False	True
13546	False	False	False	False	False	True
13547	False	False	False	False	True	True
13548	False	False	False	False	True	True

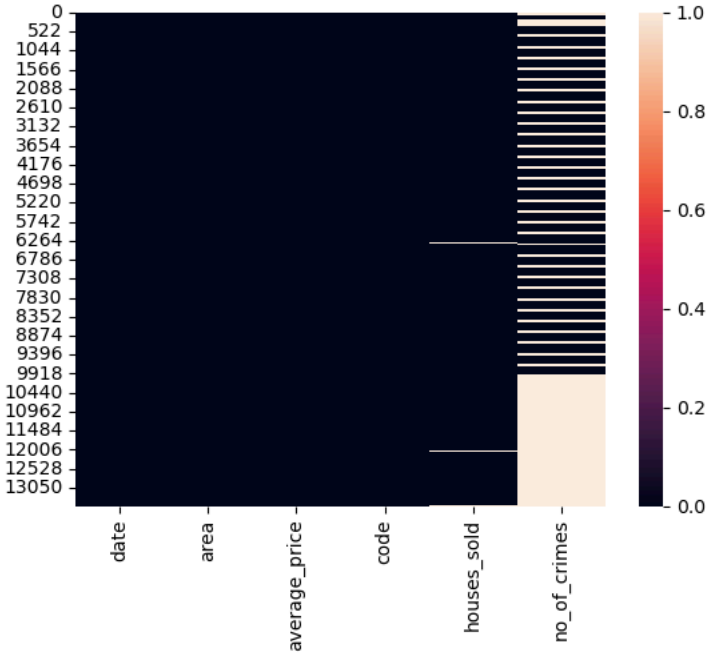
13549 rows × 6 columns

```
data.isnull().sum()
```

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

Find the null values using the heatmap.

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




```
data.dtypes
```

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

Convert the Datatypes of Date column to Datetime format.

```
data['date'] = pd.to_datetime(data.date)
```

```
data.dtypes
```

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

Create a one new column of the given dataset are 'Year'.

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

```
data.head(5)
```

	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

Create a one new column of the given dataset are 'Month' and insert a 2nd column.

```
data.insert(1,'month',data.date.dt.month)
```

```
data.head(5)
```

	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

Remove the month and year column of the given dataset.

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

```
data.head(5)
```

	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

Show all the records where 'No of Crimies' is 0. And, how many records are there?

```
data[data['no_of_crimes'] == 0]
```

	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

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

```
104
```

What is Minimum & Maximum 'average_price' per year in england?

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

```
df1 = data[data['area'] == 'england']
```

```
df1
```

	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

```
df2 = data.groupby('year').average_price.max()
```

```
df2
```

```
year
1995     200722
1996     223197
1997     265112
1998     277600
1999     354241
2000     397353
2001     451028
2002     497538
2003     488704
2004     559286
2005     555847
2006     644541
2007     830950
2008     832753
2009     782459
2010     884674
2011     959520
2012    1077366
2013    1217729
2014    1365050
2015    1353679
2016    1357231
2017    1412255
2018    1463378
2019    1294113
2020    1178166
Name: average_price, dtype: int64
```

```
df3 = data.groupby('year').average_price.min()
```

```
df3
```

```
year
1995     41688
1996     40722
1997     42353
1998     43510
1999     43969
2000     47604
2001     49045
2002     54746
2003     67520
2004     88520
2005    110454
2006    121124
2007    131175
2008    120275
2009    117079
2010    119688
2011    115328
2012    113011
2013    112008
2014    114531
2015    117156
2016    121085
2017    121858
2018    124038
2019    124567
2020    126592
Name: average_price, dtype: int64
```

What is the Maximum & Minimum No. of Crimes recorded per area ?

```
d1 = data.groupby('area').no_of_crimes.max()
```

```
d1
```

```
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
```

```
d2 = data.groupby('area').no_of_crimes.min()
```

```
d2
```

```
area
barking and dagenham      1217.0
barnet                    1703.0
bexley                     860.0
brent                     1850.0
bromley                   1441.0
camden                    2079.0
city of london               0.0
croydon                   2031.0
ealing                    1871.0
east midlands                NaN
east of england              NaN
enfield                   1635.0
england                      NaN
greenwich                 1513.0
hackney                   1870.0
hammersmith and fulham    1323.0
haringey                  1536.0
harrow                     937.0
havering                  1130.0
hillingdon                1445.0
hounslow                  1529.0
inner london                 NaN
islington                 1871.0
kensington and chelsea    1347.0
```

```
kingston upon thames       692.0
lambeth                   2381.0
lewisham                  1675.0
london                       NaN
merton                     819.0
newham                    2130.0
north east                   NaN
north west                   NaN
outer london                 NaN
redbridge                 1487.0
richmond upon thames       700.0
south east                   NaN
south west                   NaN
southwark                 2267.0
sutton                     787.0
tower hamlets             1646.0
waltham forest            1575.0
wandsworth                1582.0
west midlands                NaN
westminster               3504.0
yorks and the humber         NaN
Name: no_of_crimes, dtype: float64
```

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

```
data[data['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
```

```
data.head(5)
```

	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

Start coding or generate with AI.

Start coding or generate with AI.