

In these project , we analyses the every parameter of house in europe

In [1]: `import pandas as pd`

In [2]: `data = pd.read_csv('London Housing Data.csv')`

In [3]: `data`

Out[3]:

| | 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 [4]: `data.count()` # count a numerical value and guess null value

Out[4]:

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

In [5]: `data.isnull()` # find null value (true= Present null, False= no present null)

Out[5]:

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

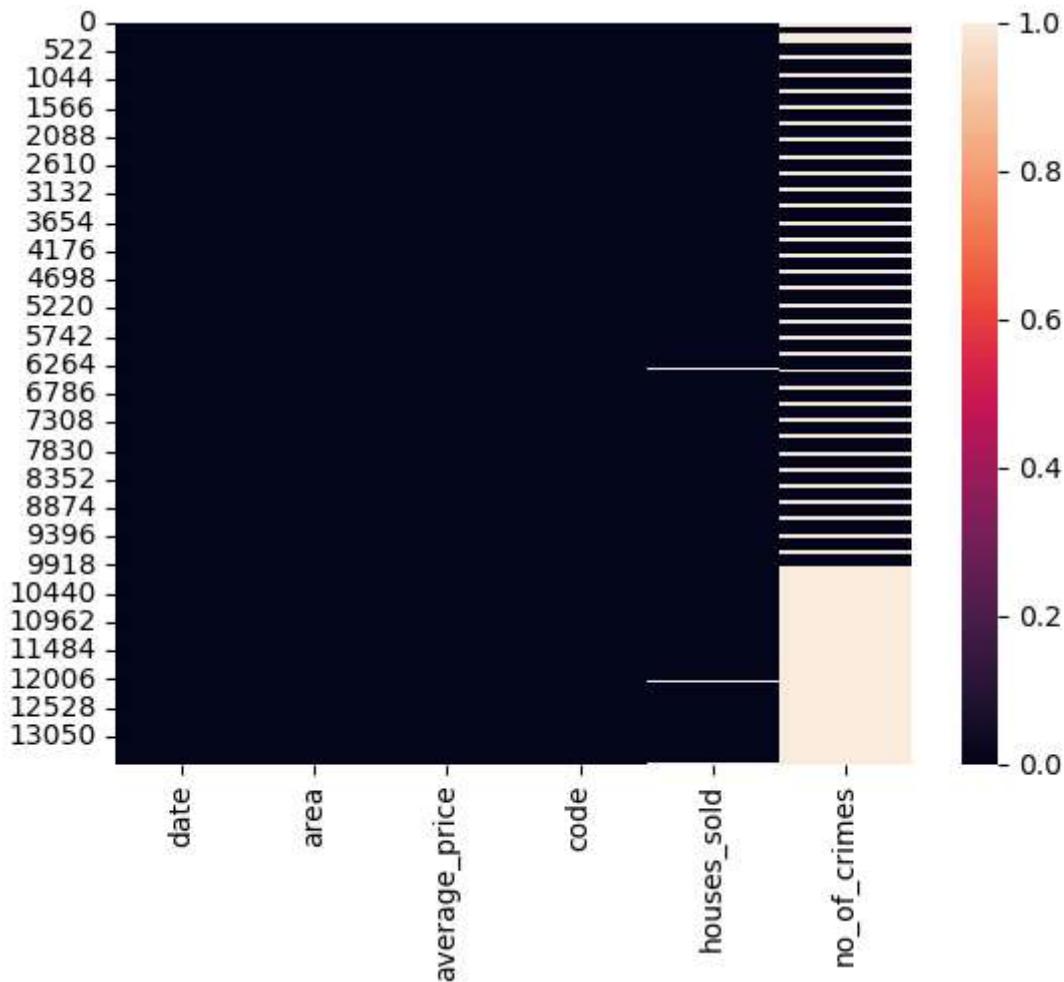
13549 rows × 6 columns

In [6]: `data.isnull().sum() # count null value in data`

Out[6]:

nullvalue plot in heatmap for better visualiazation

In [7]: `import seaborn as sns
import matplotlib.pyplot as plt`In [8]: `sns.heatmap(data.isnull())
plt.show`Out[8]: `<function matplotlib.pyplot.show(close=None, block=None)>`



date time separate

```
In [9]: data.head()
```

| | 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 |

```
In [10]: data.dtypes
```

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

```
In [11]: data.date = pd.to_datetime(data.date)
data.date
```

```
Out[11]: 0      1995-01-01
         1      1995-02-01
         2      1995-03-01
         3      1995-04-01
         4      1995-05-01
         ...
        13544  2019-09-01
        13545  2019-10-01
        13546  2019-11-01
        13547  2019-12-01
        13548  2020-01-01
Name: date, Length: 13549, dtype: datetime64[ns]
```

```
In [12]: data['year']=data.date.dt.year
data
```

| | 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

```
In [13]: data['month']= data.date.dt.month
data
```

Out[13]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year | month |
|--------------|-------------|----------------|----------------------|-------------|--------------------|---------------------|-------------|--------------|
| 0 | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 | 1 |
| 1 | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 | 2 |
| 2 | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 | 3 |
| 3 | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 | 4 |
| 4 | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 | 5 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 13544 | 2019-09-01 | england | 249942 | E92000001 | 64605.0 | NaN | 2019 | 9 |
| 13545 | 2019-10-01 | england | 249376 | E92000001 | 68677.0 | NaN | 2019 | 10 |
| 13546 | 2019-11-01 | england | 248515 | E92000001 | 67814.0 | NaN | 2019 | 11 |
| 13547 | 2019-12-01 | england | 250410 | E92000001 | NaN | NaN | 2019 | 12 |
| 13548 | 2020-01-01 | england | 247355 | E92000001 | NaN | NaN | 2020 | 1 |

13549 rows × 8 columns

FIND NO OF CRIMES

In [14]: `data.head()`

Out[14]:

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

In [15]: `data[data.no_of_crimes==0] # FIND ZERO NO OF CRIME`

Out[15]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year | month |
|------------|-------------|----------------|----------------------|-------------|--------------------|---------------------|-------------|--------------|
| 72 | 2001-01-01 | city of london | 284262 | E09000001 | 24.0 | 0.0 | 2001 | 1 |
| 73 | 2001-02-01 | city of london | 198137 | E09000001 | 37.0 | 0.0 | 2001 | 2 |
| 74 | 2001-03-01 | city of london | 189033 | E09000001 | 44.0 | 0.0 | 2001 | 3 |
| 75 | 2001-04-01 | city of london | 205494 | E09000001 | 38.0 | 0.0 | 2001 | 4 |
| 76 | 2001-05-01 | city of london | 223459 | E09000001 | 30.0 | 0.0 | 2001 | 5 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 178 | 2009-11-01 | city of london | 397909 | E09000001 | 11.0 | 0.0 | 2009 | 11 |
| 179 | 2009-12-01 | city of london | 411955 | E09000001 | 16.0 | 0.0 | 2009 | 12 |
| 180 | 2010-01-01 | city of london | 464436 | E09000001 | 20.0 | 0.0 | 2010 | 1 |
| 181 | 2010-02-01 | city of london | 490525 | E09000001 | 9.0 | 0.0 | 2010 | 2 |
| 182 | 2010-03-01 | city of london | 498241 | E09000001 | 15.0 | 0.0 | 2010 | 3 |

104 rows × 8 columns

In [16]: `len(data[data.no_of_crimes==0]) #TOTAL PLACE ZERO NO OF CRIME`

Out[16]: 104

In [17]: `data[data.no_of_crimes>1000] len(data[data.no_of_crimes>1000]) #greater than 1000 no of crimes in area`

Out[17]: 6960

In [18]: `data.max() # max no crimes`

Out[18]:

| | |
|---------------|----------------------|
| date | 2020-01-01 00:00:00 |
| area | yorks and the humber |
| average_price | 1463378 |
| code | E9200001 |
| houses_sold | 132163.0 |
| no_of_crimes | 7461.0 |
| year | 2020 |
| month | 12 |
| dtype: object | |

In [19]: `# Individual area scrapping`

In [20]: `data.head()`

Out[20]:

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

In [21]:

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

Out[21]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year | month |
|--------------|-------------|-------------|----------------------|-------------|--------------------|---------------------|-------------|--------------|
| 13248 | 1995-01-01 | england | 53203 | E92000001 | 47639.0 | NaN | 1995 | 1 |
| 13249 | 1995-02-01 | england | 53096 | E92000001 | 47880.0 | NaN | 1995 | 2 |
| 13250 | 1995-03-01 | england | 53201 | E92000001 | 67025.0 | NaN | 1995 | 3 |
| 13251 | 1995-04-01 | england | 53591 | E92000001 | 56925.0 | NaN | 1995 | 4 |
| 13252 | 1995-05-01 | england | 53678 | E92000001 | 64192.0 | NaN | 1995 | 5 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 13544 | 2019-09-01 | england | 249942 | E92000001 | 64605.0 | NaN | 2019 | 9 |
| 13545 | 2019-10-01 | england | 249376 | E92000001 | 68677.0 | NaN | 2019 | 10 |
| 13546 | 2019-11-01 | england | 248515 | E92000001 | 67814.0 | NaN | 2019 | 11 |
| 13547 | 2019-12-01 | england | 250410 | E92000001 | NaN | NaN | 2019 | 12 |
| 13548 | 2020-01-01 | england | 247355 | E92000001 | NaN | NaN | 2020 | 1 |

301 rows × 8 columns

In [22]:

```
df1.groupby('date').average_price.min() # england min no of avg price of home
```

```
Out[22]: date
1995-01-01    53203
1995-02-01    53096
1995-03-01    53201
1995-04-01    53591
1995-05-01    53678
...
2019-09-01    249942
2019-10-01    249376
2019-11-01    248515
2019-12-01    250410
2020-01-01    247355
Name: average_price, Length: 301, dtype: int64
```

```
In [23]: df1.groupby('date').average_price.min().sort_values(ascending=False) # sort by high to low
```

```
Out[23]: date
2019-12-01    250410
2019-09-01    249942
2019-08-01    249432
2019-10-01    249376
2018-08-01    248620
...
1995-10-01    52845
1995-11-01    52788
1996-03-01    52683
1996-02-01    52536
1996-01-01    52333
Name: average_price, Length: 301, dtype: int64
```

```
In [24]: df1.groupby('date').average_price.max() # max price of england
```

```
Out[24]: date
1995-01-01    53203
1995-02-01    53096
1995-03-01    53201
1995-04-01    53591
1995-05-01    53678
...
2019-09-01    249942
2019-10-01    249376
2019-11-01    248515
2019-12-01    250410
2020-01-01    247355
Name: average_price, Length: 301, dtype: int64
```

```
In [25]: df1.groupby('date').average_price.mean() # avg price of mean in england
```

```
Out[25]: date
1995-01-01    53203.0
1995-02-01    53096.0
1995-03-01    53201.0
1995-04-01    53591.0
1995-05-01    53678.0
...
2019-09-01    249942.0
2019-10-01    249376.0
2019-11-01    248515.0
2019-12-01    250410.0
2020-01-01    247355.0
Name: average_price, Length: 301, dtype: float64
```

```
In [26]: df1.groupby('date').no_of_crimes.max() # max crime in engLand
```

```
Out[26]: date
1995-01-01    NaN
1995-02-01    NaN
1995-03-01    NaN
1995-04-01    NaN
1995-05-01    NaN
...
2019-09-01    NaN
2019-10-01    NaN
2019-11-01    NaN
2019-12-01    NaN
2020-01-01    NaN
Name: no_of_crimes, Length: 301, dtype: float64
```

avg price of all city

```
In [27]: data.groupby('average_price').max() # avg price max in all city
```

| | date | area | code | houses_sold | no_of_crimes | year | month |
|---------------|------------|------------------------|-----------|-------------|--------------|------|-------|
| average_price | | | | | | | |
| 40722 | 1996-02-01 | north east | E12000001 | 2113.0 | NaN | 1996 | 2 |
| 40790 | 1996-01-01 | north east | E12000001 | 2020.0 | NaN | 1996 | 1 |
| 41588 | 1996-04-01 | north east | E12000001 | 2604.0 | NaN | 1996 | 4 |
| 41688 | 1995-10-01 | north east | E12000001 | 2674.0 | NaN | 1995 | 10 |
| 41728 | 1995-11-01 | north east | E12000001 | 2620.0 | NaN | 1995 | 11 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 1394421 | 2017-01-01 | kensington and chelsea | E09000020 | 136.0 | 1534.0 | 2017 | 1 |
| 1399839 | 2017-04-01 | kensington and chelsea | E09000020 | 124.0 | 1723.0 | 2017 | 4 |
| 1412255 | 2017-02-01 | kensington and chelsea | E09000020 | 109.0 | 1367.0 | 2017 | 2 |
| 1418032 | 2018-08-01 | kensington and chelsea | E09000020 | 103.0 | 2339.0 | 2018 | 8 |
| 1463378 | 2018-01-01 | kensington and chelsea | E09000020 | 129.0 | 1737.0 | 2018 | 1 |

13343 rows × 7 columns

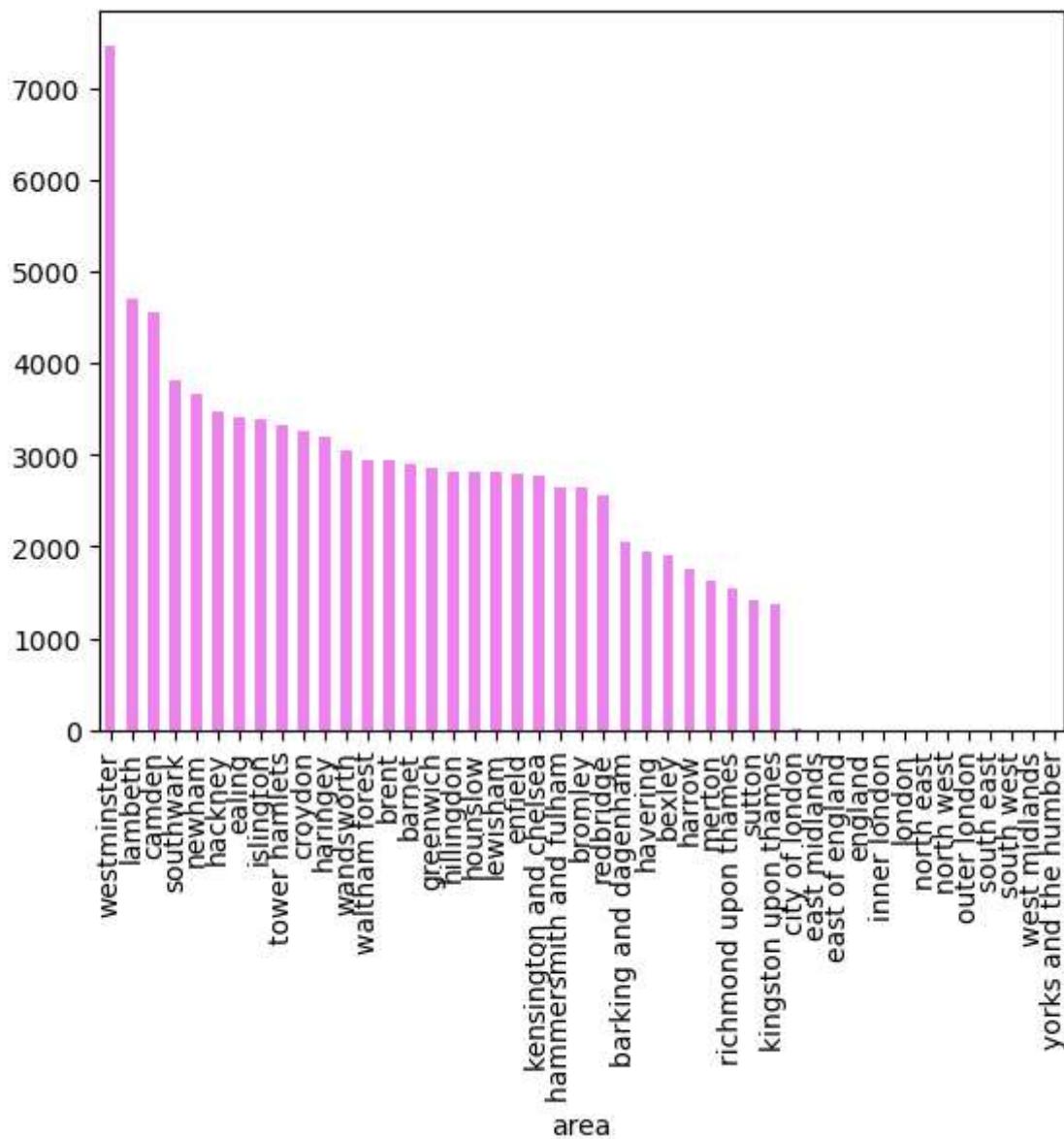
```
In [28]: data.groupby('area').no_of_crimes.max() # area wise max crime rate
```

```
Out[28]: 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 [29]: data.groupby('area').no_of_crimes.max().sort_values(ascending=False) # sort by high
```

```
Out[29]: area
westminster      7461.0
lambeth          4701.0
camden           4558.0
southwark         3821.0
newham            3668.0
hackney           3466.0
ealing             3401.0
islington         3384.0
tower hamlets     3316.0
croydon           3263.0
haringey          3199.0
wandsworth         3051.0
waltham forest     2941.0
brent              2937.0
barnet              2893.0
greenwich          2853.0
hillingdon         2819.0
hounslow            2817.0
lewisham           2813.0
enfield             2798.0
kensington and chelsea 2778.0
hammersmith and fulham 2645.0
bromley             2637.0
redbridge            2560.0
barking and dagenham 2049.0
havering             1956.0
bexley               1914.0
harrow                1763.0
merton                1623.0
richmond upon thames 1551.0
sutton                 1425.0
kingston upon thames 1379.0
city of london        10.0
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
```

```
In [30]: data.groupby('area').no_of_crimes.max().sort_values(ascending=False).plot(kind='bar', c
Out[30]: <AxesSubplot:xlabel='area'>
```



```
In [31]: data.groupby('area').no_of_crimes.min()
```

```
Out[31]: 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
```

```
In [32]: data[data.average_price>100000].area.value_counts() # avg price greater than
```

```
Out[32]:
```

| | |
|------------------------|-----|
| hammersmith and fulham | 301 |
| richmond upon thames | 301 |
| kensington and chelsea | 301 |
| camden | 301 |
| westminster | 301 |
| city of london | 290 |
| islington | 282 |
| barnet | 276 |
| wandsworth | 275 |
| kingston upon thames | 271 |
| harrow | 271 |
| inner london | 270 |
| ealing | 270 |
| haringey | 268 |
| bromley | 268 |
| merton | 266 |
| london | 262 |
| brent | 261 |
| lambeth | 260 |
| hounslow | 260 |
| hillingdon | 257 |
| outer london | 255 |
| tower hamlets | 255 |
| southwark | 253 |
| redbridge | 249 |
| hackney | 249 |
| enfield | 248 |
| sutton | 247 |
| croydon | 244 |
| south east | 243 |
| greenwich | 242 |
| havering | 241 |
| lewisham | 239 |
| waltham forest | 237 |
| bexley | 237 |
| newham | 229 |
| east of england | 225 |
| south west | 223 |
| barking and dagenham | 216 |
| england | 214 |
| west midlands | 207 |
| east midlands | 205 |
| yorks and the humber | 191 |
| north west | 190 |
| north east | 189 |

Name: area, dtype: int64

```
In [33]: # find a average price and city crime in city of London
```

```
In [34]: data.head(2)
```

```
Out[34]:
```

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

```
In [35]: london = data[data.area == 'city of london']
london
```

Out[35]:

| | date | area | average_price | code | houses_sold | no_of_crimes | year | month |
|------------|-------------|----------------|----------------------|-------------|--------------------|---------------------|-------------|--------------|
| 0 | 1995-01-01 | city of london | 91449 | E09000001 | 17.0 | NaN | 1995 | 1 |
| 1 | 1995-02-01 | city of london | 82203 | E09000001 | 7.0 | NaN | 1995 | 2 |
| 2 | 1995-03-01 | city of london | 79121 | E09000001 | 14.0 | NaN | 1995 | 3 |
| 3 | 1995-04-01 | city of london | 77101 | E09000001 | 7.0 | NaN | 1995 | 4 |
| 4 | 1995-05-01 | city of london | 84409 | E09000001 | 10.0 | NaN | 1995 | 5 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 296 | 2019-09-01 | city of london | 792034 | E09000001 | 11.0 | NaN | 2019 | 9 |
| 297 | 2019-10-01 | city of london | 827093 | E09000001 | 4.0 | NaN | 2019 | 10 |
| 298 | 2019-11-01 | city of london | 777610 | E09000001 | 5.0 | NaN | 2019 | 11 |
| 299 | 2019-12-01 | city of london | 734872 | E09000001 | NaN | NaN | 2019 | 12 |
| 300 | 2020-01-01 | city of london | 775309 | E09000001 | NaN | NaN | 2020 | 1 |

301 rows × 8 columns

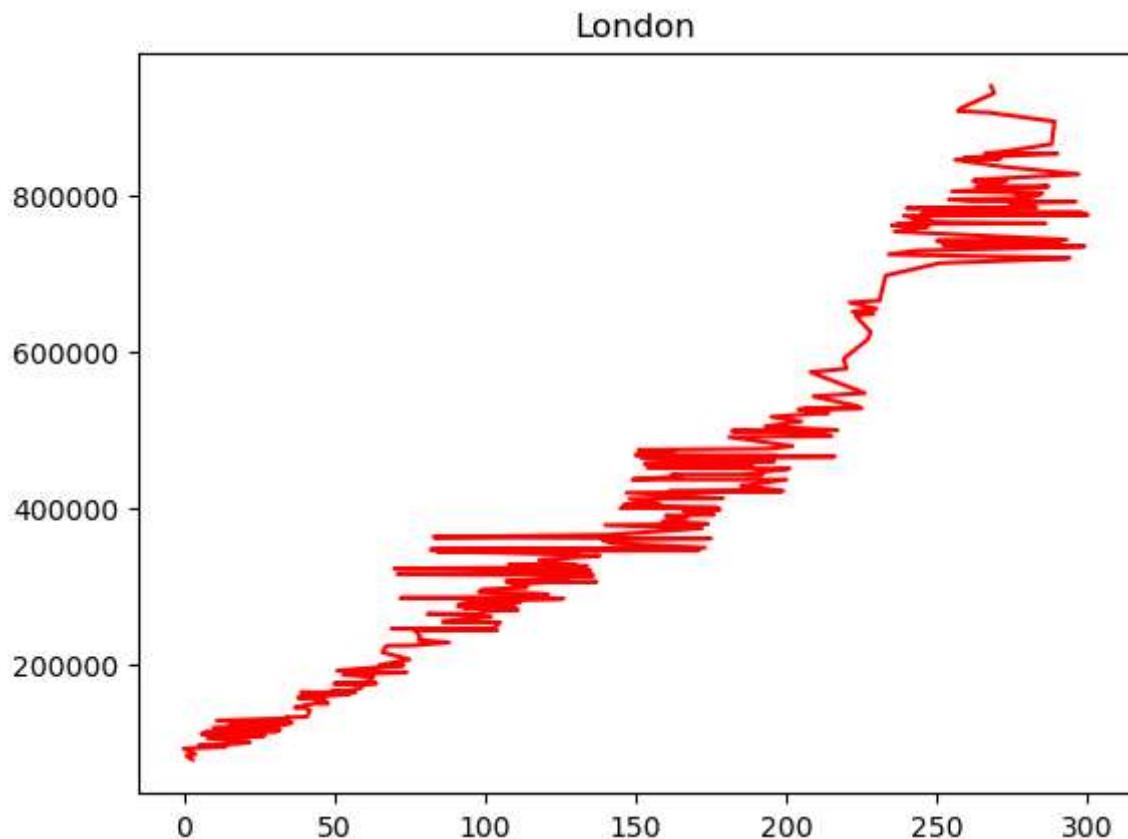
```
In [36]: london.isnull().sum() # city of London
```

```
Out[36]: date      0
area       0
average_price 0
code       0
houses_sold   2
no_of_crimes 190
year       0
month      0
dtype: int64
```

```
In [37]: london['average_price'].sort_values(ascending=False) # city of London average price
```

```
Out[37]:    268    939912  
    269    930979  
    258    911827  
    257    907965  
    267    906146  
    ...  
    0      91449  
    4      84409  
    1      82203  
    2      79121  
    3      77101  
Name: average_price, Length: 301, dtype: int64
```

```
In [38]: london['average_price'].sort_values(ascending=False).plot(kind='line',color='red',title='London')  
Out[38]: <AxesSubplot:title={'center':'London'}>
```



```
In [39]: london['average_price'].mean() # avg price mean  
Out[39]: 423942.584717608
```

```
In [40]: london['average_price'].median() # avg price median  
Out[40]: 382571.0
```

```
In [41]: london['no_of_crimes'].sort_values(ascending=False) # crimes high to low
```

```
Out[41]:    149    10.0
           151     8.0
           147     7.0
           146     7.0
           145     7.0
           ...
          296    NaN
          297    NaN
          298    NaN
          299    NaN
          300    NaN
Name: no_of_crimes, Length: 301, dtype: float64
```

```
In [42]: london['no_of_crimes'].mean()      # crimes mean
```

```
Out[42]: 0.42342342342342343
```

```
In [43]: london['no_of_crimes'].median()      # crimes median
```

```
Out[43]: 0.0
```

```
In [44]: # Find max crime which date
```

```
In [45]: data.head(2)
```

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

```
In [46]: data.groupby('date').no_of_crimes.max().sort_values(ascending=False) # 01-12-2019 crime
```

| date | no_of_crimes |
|------------|--------------|
| 2019-12-01 | 7461.0 |
| 2019-04-01 | 7227.0 |
| 2019-03-01 | 7215.0 |
| 2019-10-01 | 7208.0 |
| 2002-05-01 | 7076.0 |
| ... | |
| 2000-08-01 | NaN |
| 2000-09-01 | NaN |
| 2000-10-01 | NaN |
| 2000-11-01 | NaN |
| 2000-12-01 | NaN |

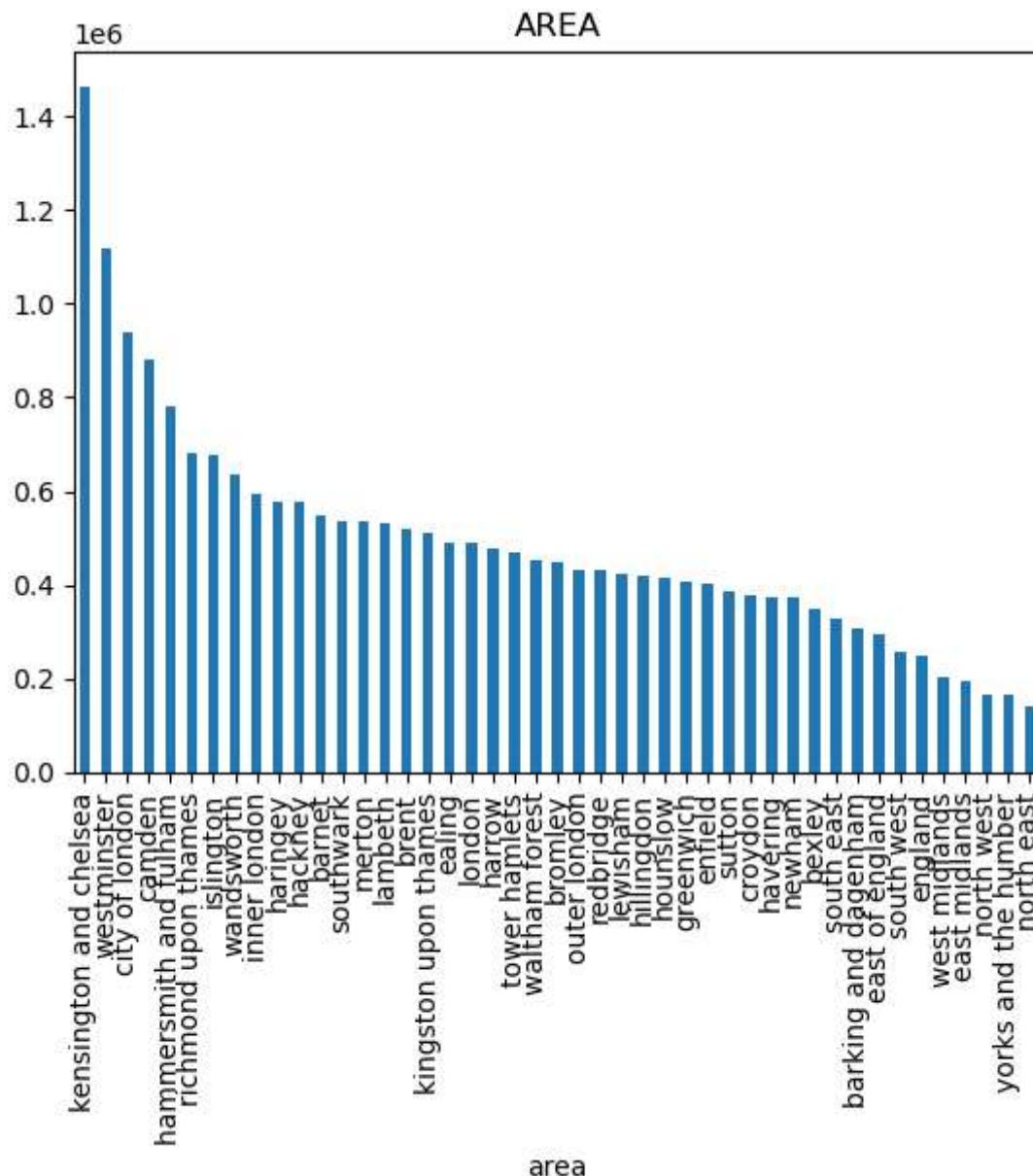
Name: no_of_crimes, Length: 301, dtype: float64

```
In [47]: #find max average price in area
```

```
In [48]: data.groupby('area').average_price.max().sort_values(ascending=False) # max avg price
```

```
Out[48]: area
kensington and chelsea    1463378
westminster                 1117408
city of london               939912
camden                      881212
hammersmith and fulham     781175
richmond upon thames        682854
islington                   677798
wandsworth                   633826
inner london                 593486
haringey                     578728
hackney                      575846
barnet                       549401
southwark                     535685
merton                        534091
lambeth                      532804
brent                         519754
kingston upon thames         511260
ealing                        489687
london                        488527
harrow                        479341
tower hamlets                 470788
waltham forest                452299
bromley                       449293
outer london                  432669
redbridge                     430708
lewisham                      423063
hillingdon                    419452
hounslow                      413672
greenwich                     408407
enfield                       404393
sutton                        387410
croydon                       378859
havering                      373762
newham                        371678
bexley                        347531
south east                     328061
barking and dagenham          305882
east of england                294035
south west                     259145
england                        250410
west midlands                  201447
east midlands                  196534
north west                     167430
yorks and the humber           166932
north east                     139400
Name: average_price, dtype: int64
```

```
In [49]: data.groupby('area').average_price.max().sort_values(ascending=False).plot(kind='bar',
Out[49]: <AxesSubplot:title={'center':'AREA'}, xlabel='area'>
```



In [50]: `#find no house sold which date`

In [51]: `data.head(2)`

Out[51]:

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

In [52]: `data.groupby('date').houses_sold.max().sort_values(ascending=False) # max house so`

```
Out[52]: date
2002-05-01    132163.0
2004-07-01    126778.0
2002-07-01    126475.0
2002-08-01    126360.0
2001-08-01    124849.0
...
2010-01-01    34383.0
2009-02-01    26660.0
2009-01-01    25782.0
2019-12-01      NaN
2020-01-01      NaN
Name: houses_sold, Length: 301, dtype: float64
```

```
In [53]: # find area code
```

```
In [54]: data.head()
```

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

```
In [55]: df2=data[['area','code']]
df2
```

Out[55]:

| | area | code |
|--------------|----------------|-----------|
| 0 | city of london | E09000001 |
| 1 | city of london | E09000001 |
| 2 | city of london | E09000001 |
| 3 | city of london | E09000001 |
| 4 | city of london | E09000001 |
| ... | ... | ... |
| 13544 | england | E92000001 |
| 13545 | england | E92000001 |
| 13546 | england | E92000001 |
| 13547 | england | E92000001 |
| 13548 | england | E92000001 |

13549 rows × 2 columns

In [56]:

```
df2[df2.duplicated()]
df2
```

Out[56]:

| | area | code |
|--------------|----------------|-----------|
| 0 | city of london | E09000001 |
| 1 | city of london | E09000001 |
| 2 | city of london | E09000001 |
| 3 | city of london | E09000001 |
| 4 | city of london | E09000001 |
| ... | ... | ... |
| 13544 | england | E92000001 |
| 13545 | england | E92000001 |
| 13546 | england | E92000001 |
| 13547 | england | E92000001 |
| 13548 | england | E92000001 |

13549 rows × 2 columns

In [57]:

```
df2.drop_duplicates()
```

Out[57]:

| | area | code |
|------|------------------------|-----------|
| 0 | city of london | E09000001 |
| 301 | barking and dagenham | E09000002 |
| 602 | barnet | E09000003 |
| 903 | bexley | E09000004 |
| 1204 | brent | E09000005 |
| 1505 | bromley | E09000006 |
| 1806 | camden | E09000007 |
| 2107 | croydon | E09000008 |
| 2408 | ealing | E09000009 |
| 2709 | enfield | E09000010 |
| 2723 | enfield | E09000030 |
| 2724 | tower hamlets | E09000010 |
| 2725 | tower hamlets | E09000030 |
| 3013 | greenwich | E09000011 |
| 3314 | hackney | E09000012 |
| 3354 | hackney | E12000008 |
| 3355 | south east | E09000012 |
| 3356 | south east | E12000008 |
| 3618 | hammersmith and fulham | E09000013 |
| 3919 | haringey | E09000014 |
| 4220 | harrow | E09000015 |
| 4521 | havering | E09000016 |
| 4822 | hillingdon | E09000017 |
| 5123 | hounslow | E09000018 |
| 5424 | islington | E09000019 |
| 5725 | kensington and chelsea | E09000020 |
| 6026 | kingston upon thames | E09000021 |
| 6327 | lambeth | E09000022 |
| 6628 | lewisham | E09000023 |
| 6929 | merton | E09000024 |
| 7230 | newham | E09000025 |
| 7531 | redbridge | E09000026 |
| 7832 | richmond upon thames | E09000027 |

| | area | code |
|--------------|----------------------|-----------|
| 8133 | southwark | E09000028 |
| 8434 | sutton | E09000029 |
| 9035 | waltham forest | E09000031 |
| 9336 | wandsworth | E09000032 |
| 9637 | westminster | E09000033 |
| 9938 | inner london | E13000001 |
| 10239 | outer london | E13000002 |
| 10540 | north east | E12000001 |
| 10841 | north west | E12000002 |
| 11142 | yorks and the humber | E12000003 |
| 11443 | east midlands | E12000004 |
| 11744 | west midlands | E12000005 |
| 12045 | east of england | E12000006 |
| 12346 | london | E12000007 |
| 12947 | south west | E12000009 |
| 13248 | england | E92000001 |

In []: