In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as pt
import seaborn as sb
```

In [2]:
```
df = pd.read_csv(r"C:\Users\mousm\Downloads\bia-californa-housing-price-prediciton\houseprices.csv")
```

In [3]:
```
df
```

Out[3]:

	id	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude
0	37137	1.7062	35.0	4.966368	1.096539	1318.0	2.844411	39.75	-121.85
1	37138	1.3882	22.0	4.187035	1.098229	2296.0	3.180218	33.95	-118.29
2	37139	7.7197	21.0	7.129436	0.959276	1535.0	2.888889	33.61	-117.81
3	37140	4.6806	49.0	4.769697	1.048485	707.0	1.743590	34.17	-118.34
4	37141	3.1284	25.0	3.765306	1.081633	4716.0	2.003827	34.17	-118.29
...	...	...	...	...	...	...	...	...	...
24754	61891	2.2875	34.0	3.914729	1.085271	866.0	2.071429	34.44	-119.75
24755	61892	3.0781	33.0	4.771971	1.038674	1628.0	2.326848	34.09	-117.96
24756	61893	2.6961	14.0	4.593960	1.170380	3900.0	2.540034	37.51	-120.83
24757	61894	7.2315	8.0	7.508403	1.018692	1388.0	2.601202	33.67	-117.98
24758	61895	5.7260	30.0	6.000000	1.000000	15.0	2.500000	37.96	-122.47

24759 rows × 9 columns

In [4]:
```
# Get the basic information about the dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24759 entries, 0 to 24758
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   id          24759 non-null  int64  
 1   MedInc      24759 non-null  float64
 2   HouseAge    24759 non-null  float64
 3   AveRooms    24759 non-null  float64
 4   AveBedrms   24759 non-null  float64
 5   Population  24759 non-null  float64
 6   AveOccup    24759 non-null  float64
 7   Latitude    24759 non-null  float64
 8   Longitude   24759 non-null  float64
dtypes: float64(8), int64(1)
memory usage: 1.7 MB
```

In [5]:
```
# Get the number of rows and columns in the dataset
df.shape
```

Out[5]:
```
(24759, 9)
```

In [6]:
```
# Get the data types of each column in the dataset
df.dtypes
```

Out[6]:
```
id              int64
MedInc        float64
HouseAge      float64
AveRooms      float64
AveBedrms     float64
Population    float64
AveOccup      float64
Latitude      float64
Longitude     float64
dtype: object
```

In [7]:
```
# Check for missing values in the dataset
df.isnull().sum()
```

Out[7]:
```
id            0
MedInc        0
HouseAge      0
AveRooms      0
AveBedrms     0
Population    0
AveOccup      0
Latitude      0
Longitude     0
dtype: int64
```

In [8]:
```
# Get the correlation between each pair of columns in the dataset
df.corr()
```

Out[8]:

	id	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude
id	1.000000	0.007094	0.012275	-0.001345	-0.000858	0.000483	0.000043	0.002873	-0.004015
MedInc	0.007094	1.000000	-0.099300	0.675771	-0.106945	-0.011196	-0.010099	-0.049784	-0.053319
HouseAge	0.012275	-0.099300	1.000000	-0.165603	-0.060423	-0.242463	0.011319	0.014746	-0.081146
AveRooms	-0.001345	0.675771	-0.165603	1.000000	0.301382	-0.033811	0.024040	0.104201	-0.091081
AveBedrms	-0.000858	-0.106945	-0.060423	0.301382	1.000000	-0.005116	-0.027031	0.025227	0.021194
Population	0.000483	-0.011196	-0.242463	-0.033811	-0.005116	1.000000	0.041995	-0.070709	0.076619
AveOccup	0.000043	-0.010099	0.011319	0.024040	-0.027031	0.041995	1.000000	-0.049545	0.058530
Latitude	0.002873	-0.049784	0.014746	0.104201	0.025227	-0.070709	-0.049545	1.000000	-0.936069
Longitude	-0.004015	-0.053319	-0.081146	-0.091081	0.021194	0.076619	0.058530	-0.936069	1.000000

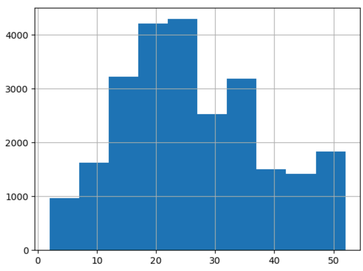
In [9]:
```
df.describe()
```

Out[9]:

	id	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude
count	24759.000000	24759.000000	24759.000000	24759.000000	24759.000000	24759.000000	24759.000000	24759.000000	24759.000000
mean	49516.000000	3.832618	26.041561	5.168789	1.063599	1679.327548	2.830808	35.598007	-119.570208
std	7147.451994	1.797503	12.177907	1.252874	0.123630	1365.598976	1.615320	2.097787	1.977200
min	37137.000000	0.499900	2.000000	1.000000	0.560000	3.000000	0.764706	32.550000	-124.230000
25%	43326.500000	2.590150	17.000000	4.356443	1.020460	955.000000	2.400000	33.930000	-121.800000
50%	49516.000000	3.504600	25.000000	5.077143	1.054094	1398.000000	2.751592	34.200000	-118.460000
75%	55705.500000	4.687500	35.000000	5.858646	1.088295	1874.000000	3.129167	37.720000	-118.020000
max	61895.000000	15.000100	52.000000	56.269231	10.500000	35682.000000	230.172414	41.950000	-114.550000

In [10]:
```
df['HouseAge'].hist()
```

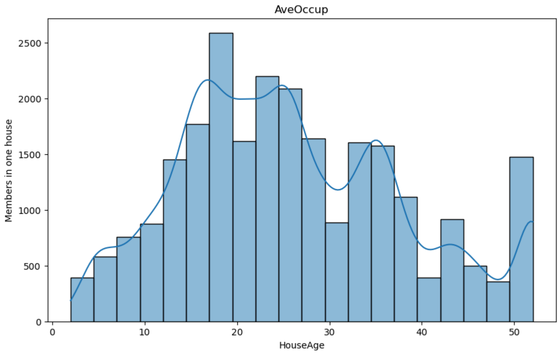
Out[10]:
```
<Axes: >
```



In [ ]:
```
# Create a line plot of the average house price per square foot over time
df.plot(x='Population', y='AveRooms', kind='line')
pt.title('Average rooms wrt population')
pt.xlabel('Population')
pt.ylabel('Average Rooms')
# Rotate x-axis labels for better readability (if needed)
pt.xticks(rotation=45)
pt.show()
```

In [11]:
```
# Data visualization
# Histogram of Amount
pt.figure(figsize=(10, 6))
sb.histplot(df['HouseAge'], bins=20, kde=True)
pt.title('AveOccup')
pt.xlabel('HouseAge')
pt.ylabel('Members in one house')
pt.show()
```

```
C:\Users\mousm\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



In [ ]:
```
sb.boxplot(
    x = "AveRooms",
    y = "Population",
    showmeans=True,
```

```
    data=df
)
```

In [ ]:

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as pt
import seaborn as sb
```

In [2]:
```
df = pd.read_csv(r"C:\Users\mousm\Downloads\bia-californa-housing-price-prediciton\houseprices.csv")
```

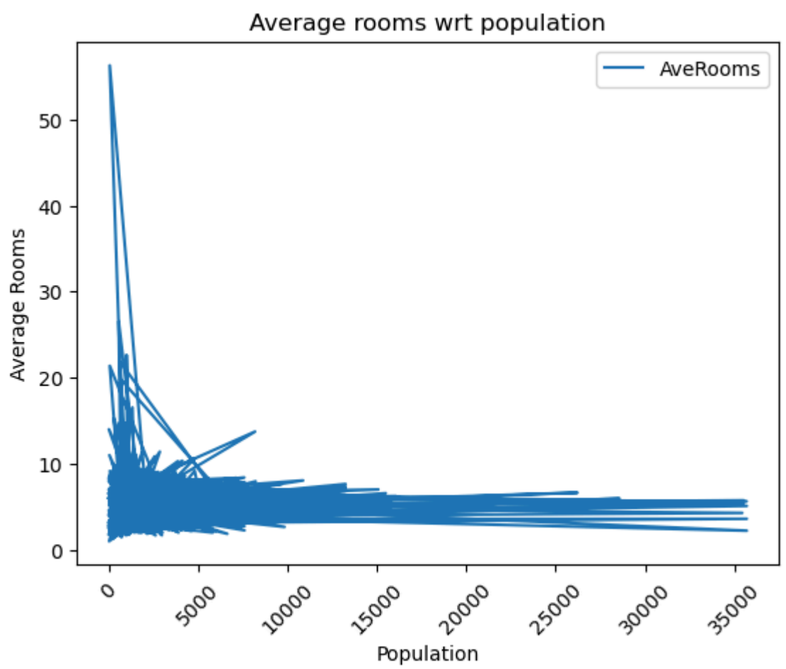
In [3]:
```
df
```

Out[3]:

	id	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude
0	37137	1.7062	35.0	4.966368	1.096539	1318.0	2.844411	39.75	-121.85
1	37138	1.3882	22.0	4.187035	1.098229	2296.0	3.180218	33.95	-118.29
2	37139	7.7197	21.0	7.129436	0.959276	1535.0	2.888889	33.61	-117.81
3	37140	4.6806	49.0	4.769697	1.048485	707.0	1.743590	34.17	-118.34
4	37141	3.1284	25.0	3.765306	1.081633	4716.0	2.003827	34.17	-118.29
...	...	...	...	...	...	...	...	...	...
24754	61891	2.2875	34.0	3.914729	1.085271	866.0	2.071429	34.44	-119.75
24755	61892	3.0781	33.0	4.771971	1.038674	1628.0	2.326848	34.09	-117.96
24756	61893	2.6961	14.0	4.593960	1.170380	3900.0	2.540034	37.51	-120.83
24757	61894	7.2315	8.0	7.508403	1.018692	1388.0	2.601202	33.67	-117.98
24758	61895	5.7260	30.0	6.000000	1.000000	15.0	2.500000	37.96	-122.47

24759 rows × 9 columns

In [4]:
```
# Create a line plot of the average house price per square foot over time
df.plot(x='Population', y='AveRooms', kind='line')
pt.title('Average rooms wrt population')
pt.xlabel('Population')
pt.ylabel('Average Rooms')
# Rotate x-axis labels for better readability (if needed)
pt.xticks(rotation=45)
pt.show()
```



In [ ]:
```
sb.boxplot(
    x = "AveRooms",
    y = "Population",
    showmeans=True,
    data=df
)
```

Out[ ]: <Axes: xlabel='AveRooms', ylabel='Population'>

In [ ]: