```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
data=pd.read_csv("/content/householdtask3.csv")
```

```python
display(data.head(10))
```

	year	tot_hhs	own	own_wm	own_prop	own_wm_prop	prop_hhs	age	size	income
0	2008	1560859	1087580	574406	69.7	36.8	100.0	35.9	2.7	46704
1	2008	185965	71256	39405	38.3	21.2	11.9	29.9	2.6	23404
2	2008	312376	191470	48424	61.3	15.5	20.0	40.0	2.3	16747
3	2008	312333	196203	84171	62.8	26.9	20.0	34.7	2.8	31308
4	2008	312240	217657	141318	69.7	45.3	20.0	31.5	3.0	49106
5	2008	312336	229014	147658	73.3	47.3	20.0	35.3	2.6	61674
6	2008	311574	253235	152835	81.3	49.1	20.0	39.3	2.5	96861
7	2008	312761	194358	49448	62.1	15.8	20.0	38.7	2.5	23680
8	2008	311973	206342	86390	66.1	27.7	20.0	36.1	2.7	34155
9	2008	311840	194361	108065	62.3	34.7	20.0	33.0	2.8	49771

```python
display(data.tail(8))
```

	year	tot_hhs	own	own_wm	own_prop	own_wm_prop	prop_hhs	age	size	income
62	2020	350840	282193	170510	80.4	48.6	20.0	39.8	2.7	146672
63	2020	352137	182056	45300	51.7	12.9	20.0	40.6	2.5	33200
64	2020	350530	198616	80783	56.7	23.0	20.0	37.4	2.8	51756
65	2020	352564	213893	119637	60.7	33.9	20.1	36.9	2.8	69779
66	2020	350182	235256	141104	67.2	40.3	19.9	35.0	3.0	88944
67	2020	351328	288779	187838	82.2	53.5	20.0	39.6	2.6	104277
68	2020	329588	156459	107753	47.5	32.7	18.8	31.1	3.2	69581
69	2020	388013	314154	38270	81.0	9.9	22.1	69.8	1.7	34712

```python
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 70 entries, 0 to 69
Data columns (total 13 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   year         70 non-null     int64
 1   tot_hhs      70 non-null     int64
 2   own          70 non-null     int64
 3   own_wm       70 non-null     int64
 4   own_prop     70 non-null     float64
 5   own_wm_prop  70 non-null     float64
 6   prop_hhs     70 non-null     float64
 7   age          70 non-null     float64
 8   size         70 non-null     float64
 9   income       70 non-null     int64
 10  expenditure  70 non-null     int64
 11  eqv_income   70 non-null     int64
 12  eqv_exp      70 non-null     int64
dtypes: float64(5), int64(8)
memory usage: 7.2 KB
```

```python
data.shape
```

```
(70, 13)
```

```python
data.describe()
```

|	year	tot_hhs	own	own_wm	own_prop	own_wm_prop
count	70.000000	7.000000e+01	7.000000e+01	70.000000	70.000000	70.000000
mean	2014.000000	4.125193e+05	2.689711e+05	129639.457143	63.505714	30.348571
std	4.273274	3.491547e+05	2.385230e+05	128627.124758	15.145257	14.010004
min	2008.000000	1.322150e+05	3.008000e+04	14220.000000	22.800000	5.100000
25%	2011.000000	3.123460e+05	1.838408e+05	50636.750000	55.400000	15.850000
50%	2014.000000	3.276340e+05	2.153065e+05	107909.000000	64.200000	32.650000
75%	2017.000000	3.468958e+05	2.604148e+05	147563.000000	73.675000	42.050000
max	2020.000000	1.756740e+06	1.125000e+06	574662.000000	88.100000	58.200000

Data visualization

```
#Scatter plot

# Define the desired figure size
plt.figure(figsize=(8,4))

#create a scatter plot
plt.scatter(data['year'],data['own'],c='green',s=100)

#setting x and y labels
plt.xlabel("year")
plt.ylabel("own")

#adding title to the plot
plt.title("Scatter Plot of ownership over years")

# Display the plot
plt.show()
```
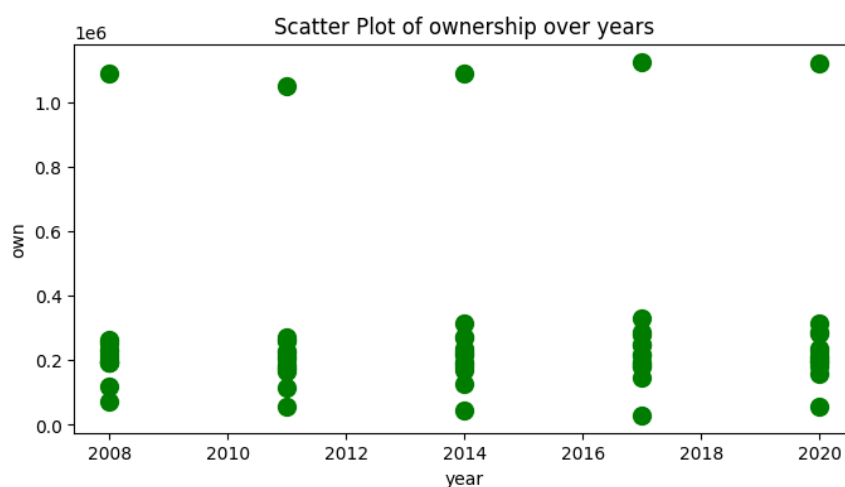


```
#Line chart

# Define the desired figure size
plt.figure(figsize=(8,4))

#create a line chart
plt.plot(data['year'],data['own'],color='purple',marker='s',linestyle='--')

#setting x and y labels
plt.xlabel("year")
plt.ylabel("own")

#adding title to the plot
plt.title("Line chart with year against own")

plt.grid(True)

# Display the plot
plt.show()
```
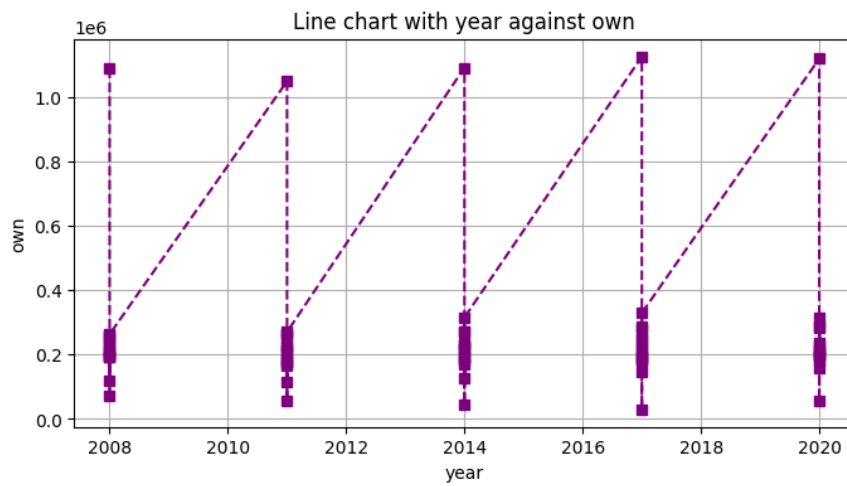
Line chart with year against own

#Histogram

```python
# Define the desired figure size
plt.figure(figsize=(8,3))

#create a histogram
plt.hist(data['income'],bins=10,edgecolor='black')

#setting x and y labels
plt.xlabel("Income")
plt.ylabel("Frequency")

#adding title to the plot
plt.title("Histogram of Income")

# Display the plot
plt.show()
```
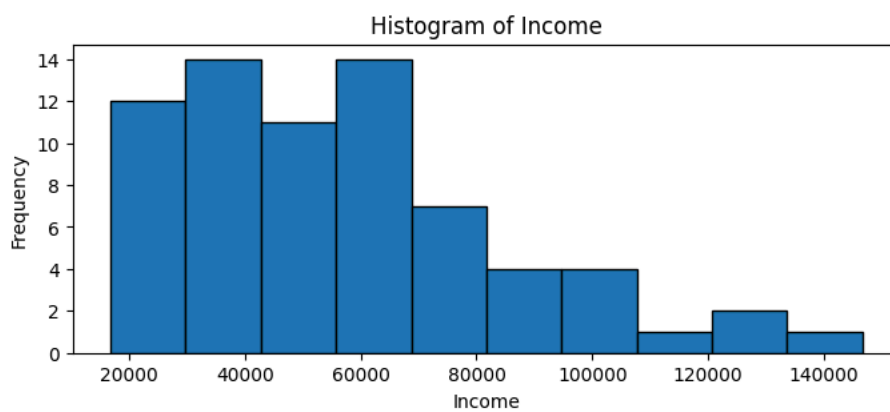
```
#Bar chart

# Define the desired figure size
plt.figure(figsize=(8,4))

#create a bar chart
plt.bar(data['year'],data['own'],width=2,color='skyblue',label='ownership')

#setting x and y labels
plt.xlabel("year")
plt.ylabel("own")

#adding title to the plot
plt.title("Bar chart of ownership over years")
plt.legend()

# Display the plot
plt.show()
```