

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

```
In [2]: income_df=pd.read_csv(r'C:\Users\archa\Downloads\Inc_Exp_Data.csv')  
income_df
```

Out[2]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_I
0	5000	8000	3	2000	
1	6000	7000	2	3000	
2	10000	4500	2	0	
3	10000	2000	1	0	
4	12500	12000	2	3000	
5	14000	8000	2	0	
6	15000	16000	3	35000	
7	18000	20000	5	8000	
8	19000	9000	2	0	
9	20000	9000	4	0	
10	20000	18000	4	8000	
11	22000	25000	6	12000	
12	23400	5000	3	0	
13	24000	10500	6	0	
14	24000	10000	4	0	
15	25000	12300	3	0	
16	25000	20000	3	3500	
17	25000	10000	6	0	
18	29000	6600	2	2000	
19	30000	13000	4	0	
20	30500	25000	5	5000	
21	32000	15000	4	0	
22	34000	19000	6	0	
23	34000	25000	3	4000	
24	35000	12000	3	0	
25	35000	25000	4	0	
26	39000	8000	4	0	
27	40000	10000	4	0	
28	42000	15000	4	0	
29	43000	12000	4	0	

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_HH_Income
30	45000	25000	6	0	45000
31	45000	40000	6	3500	45000
32	45000	10000	2	1000	45000
33	45000	22000	4	2500	45000
34	46000	25000	5	3500	46000
35	47000	15000	7	0	47000
36	50000	20000	4	0	50000
37	50500	20000	3	0	50500
38	55000	45000	6	12000	55000
39	60000	10000	3	0	60000
40	60000	50000	6	10000	60000
41	65000	20000	4	5000	65000
42	70000	9000	2	0	70000
43	80000	20000	4	0	80000
44	85000	25000	5	0	85000
45	90000	48000	7	0	90000
46	98000	25000	5	0	98000
47	100000	30000	6	0	100000
48	100000	50000	4	20000	100000
49	100000	40000	6	10000	100000

```
In [3]: income_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 7 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Mthly_HH_Income                      50 non-null    int64
1   Mthly_HH_Expense                     50 non-null    int64
2   No_of_Fly_Members                    50 non-null    int64
3   Emi_or_Rent_Amt                      50 non-null    int64
4   Annual_HH_Income                     50 non-null    int64
5   Highest_Qualified_Member             50 non-null    object
6   No_of_Earning_Members                50 non-null    int64
dtypes: int64(6), object(1)
memory usage: 2.9+ KB
```

In [4]: `income_df.shape`

Out[4]: (50, 7)


In [5]: `income_df.columns`

Out[5]: Index(['Mthly_HH_Income', 'Mthly_HH_Expense', 'No_of_Fly_Members',
'Emi_or_Rent_Amt', 'Annual_HH_Income', 'Highest_Qualified_Member',
'No_of_Earning_Members'],
dtype='object')

In [6]: `income_df.describe()`

Out[6]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_HH_Income
count	50.000000	50.000000	50.000000	50.000000	50.000000
mean	41558.000000	18818.000000	4.060000	3060.000000	35000.000000
std	26097.908979	12090.216824	1.517382	6241.434948	15000.000000
min	5000.000000	2000.000000	1.000000	0.000000	5000.000000
25%	23550.000000	10000.000000	3.000000	0.000000	15000.000000
50%	35000.000000	15500.000000	4.000000	0.000000	25000.000000
75%	50375.000000	25000.000000	5.000000	3500.000000	35000.000000
max	100000.000000	50000.000000	7.000000	35000.000000	50000.000000

◀  ▶

In [7]: `income_df.isna()`

Out[7]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_I
0	False	False	False	False	
1	False	False	False	False	
2	False	False	False	False	
3	False	False	False	False	
4	False	False	False	False	
5	False	False	False	False	
6	False	False	False	False	
7	False	False	False	False	
8	False	False	False	False	
9	False	False	False	False	
10	False	False	False	False	
11	False	False	False	False	
12	False	False	False	False	
13	False	False	False	False	
14	False	False	False	False	
15	False	False	False	False	
16	False	False	False	False	
17	False	False	False	False	
18	False	False	False	False	
19	False	False	False	False	
20	False	False	False	False	
21	False	False	False	False	
22	False	False	False	False	
23	False	False	False	False	
24	False	False	False	False	
25	False	False	False	False	
26	False	False	False	False	
27	False	False	False	False	
28	False	False	False	False	
29	False	False	False	False	

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_I
30	False	False	False	False	
31	False	False	False	False	
32	False	False	False	False	
33	False	False	False	False	
34	False	False	False	False	
35	False	False	False	False	
36	False	False	False	False	
37	False	False	False	False	
38	False	False	False	False	
39	False	False	False	False	
40	False	False	False	False	
41	False	False	False	False	
42	False	False	False	False	
43	False	False	False	False	
44	False	False	False	False	
45	False	False	False	False	
46	False	False	False	False	
47	False	False	False	False	
48	False	False	False	False	
49	False	False	False	False	

```
In [8]: income_df['Mthly_HH_Expense'].mean()
```

```
Out[8]: np.float64(18818.0)
```

```
In [9]: income_df['Mthly_HH_Expense'].median()
```

```
Out[9]: 15500.0
```

```
In [10]: income_df['Mthly_HH_Expense'].mode()
```

```
Out[10]: 0    25000
         Name: Mthly_HH_Expense, dtype: int64
```

```
In [11]: mth_exp_temp=pd.crosstab(index=income_df['Mthly_HH_Expense'],columns='count')
         mth_exp_temp.reset_index(inplace=True)
         mth_exp_temp[mth_exp_temp['count']==income_df.Mthly_HH_Expense.value_counts().max()]
```

Out[11]:

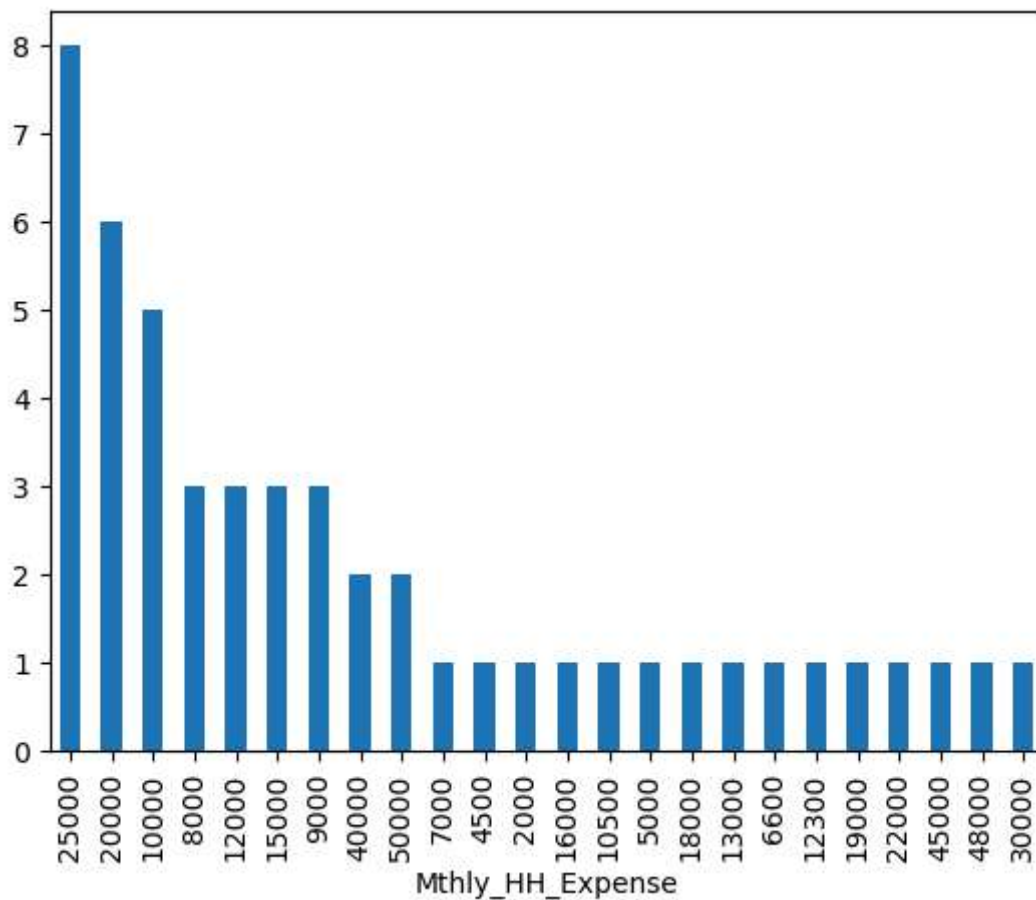
col_0	Mthly_HH_Expense	count
18	25000	8

In [12]: `income_df.columns`

Out[12]: Index(['Mthly_HH_Income', 'Mthly_HH_Expense', 'No_of_Fly_Members', 'Emi_or_Rent_Amt', 'Annual_HH_Income', 'Highest_Qualified_Member', 'No_of_Earning_Members'], dtype='object')

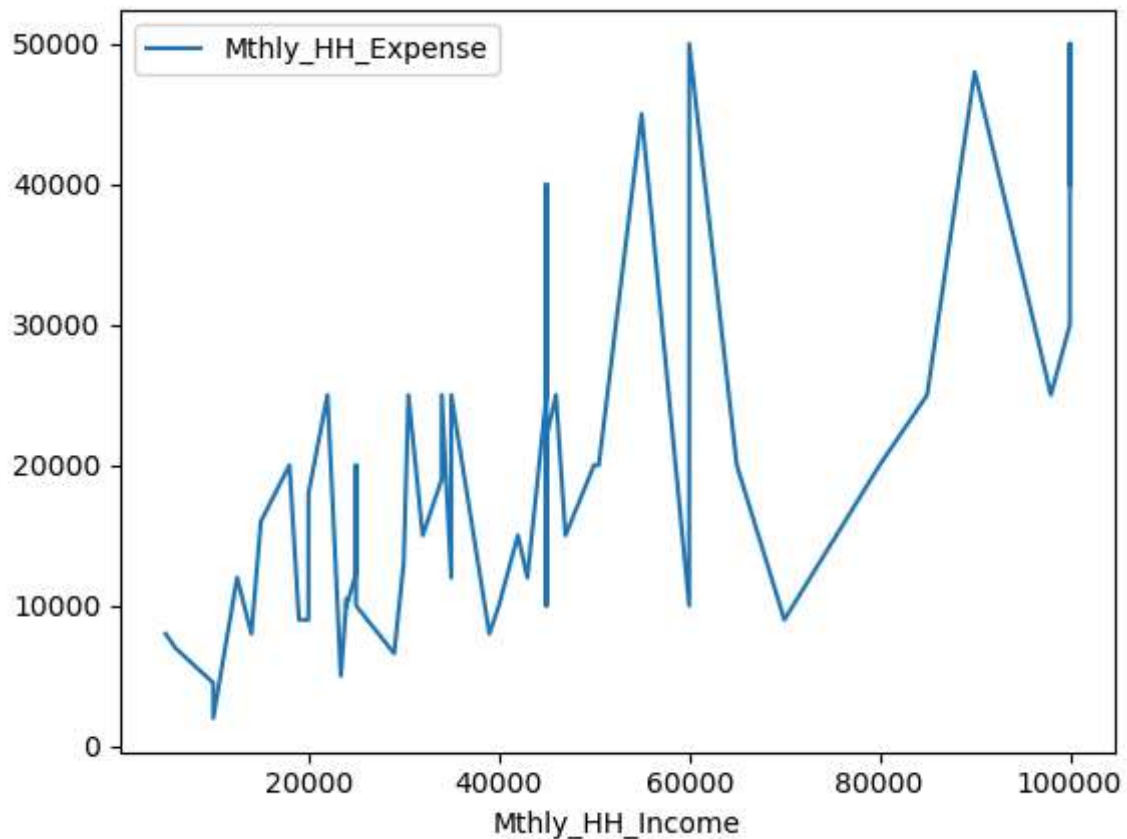
In [20]: `income_df['Mthly_HH_Expense'].value_counts().plot(kind='bar')`

Out[20]: <Axes: xlabel='Mthly_HH_Expense'>



In [21]: `income_df.plot(x='Mthly_HH_Income', y='Mthly_HH_Expense')`

Out[21]: <Axes: xlabel='Mthly_HH_Income'>



In []:

```
In [23]: IQR=income_df['Mthly_HH_Expense'].quantile(0.75)-income_df['Mthly_HH_Expense'].quantile(0.25)
```

Out[23]: np.float64(15000.0)

```
In [25]: income_df['Mthly_HH_Expense'].var()
```

Out[25]: 146173342.85714287

```
In [26]: income_df['Mthly_HH_Expense'].std()
```

Out[26]: 12090.216824240286

```
In [27]: coef=income_df['Mthly_HH_Expense'].std()/income_df['Mthly_HH_Expense'].mean()
coef
```

Out[27]: np.float64(0.6424814977277227)

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: