03 June 2025

Load the file

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

income_df = pd.read_csv(r"C:\Users\Windows10 Pro\Downloads\DataScience_AI\2025\Jun2
```

In [2]:
```
income_df
```

Out[2]:

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

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

Out[3]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annual_HH
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	

In [4]:
```
# Analyze the data
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 [5]:
```
income_df.shape
```

Out[5]: (50, 7)

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

Out[6]:

	count	mean	std	min	25%	50%	
Mthly_HH_Income	50.0	41558.00	26097.908979	5000.0	23550.0	35000.0	50
Mthly_HH_Expense	50.0	18818.00	12090.216824	2000.0	10000.0	15500.0	25
No_of_Fly_Members	50.0	4.06	1.517382	1.0	3.0	4.0	
Emi_or_Rent_Amt	50.0	3060.00	6241.434948	0.0	0.0	0.0	3
Annual_HH_Income	50.0	490019.04	320135.792123	64200.0	258750.0	447420.0	594
No_of_Earning_Members	50.0	1.46	0.734291	1.0	1.0	1.0	

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

Out[7]:
```
Mthly_HH_Income           False
Mthly_HH_Expense          False
No_of_Fly_Members         False
Emi_or_Rent_Amt           False
Annual_HH_Income          False
Highest_Qualified_Member  False
No_of_Earning_Members     False
dtype: bool
```

What is Mean Expense of a Household?

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

Out[8]: 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
```

What is Monthly Expense for most of the Households?

In [12]:
```
mth_exp_tmp = pd.crosstab(index=income_df["Mthly_HH_Expense"], columns="count")
mth_exp_tmp.reset_index(inplace=True)
mth_exp_tmp[mth_exp_tmp['count'] == income_df.Mthly_HH_Expense.value_counts().max()]
```

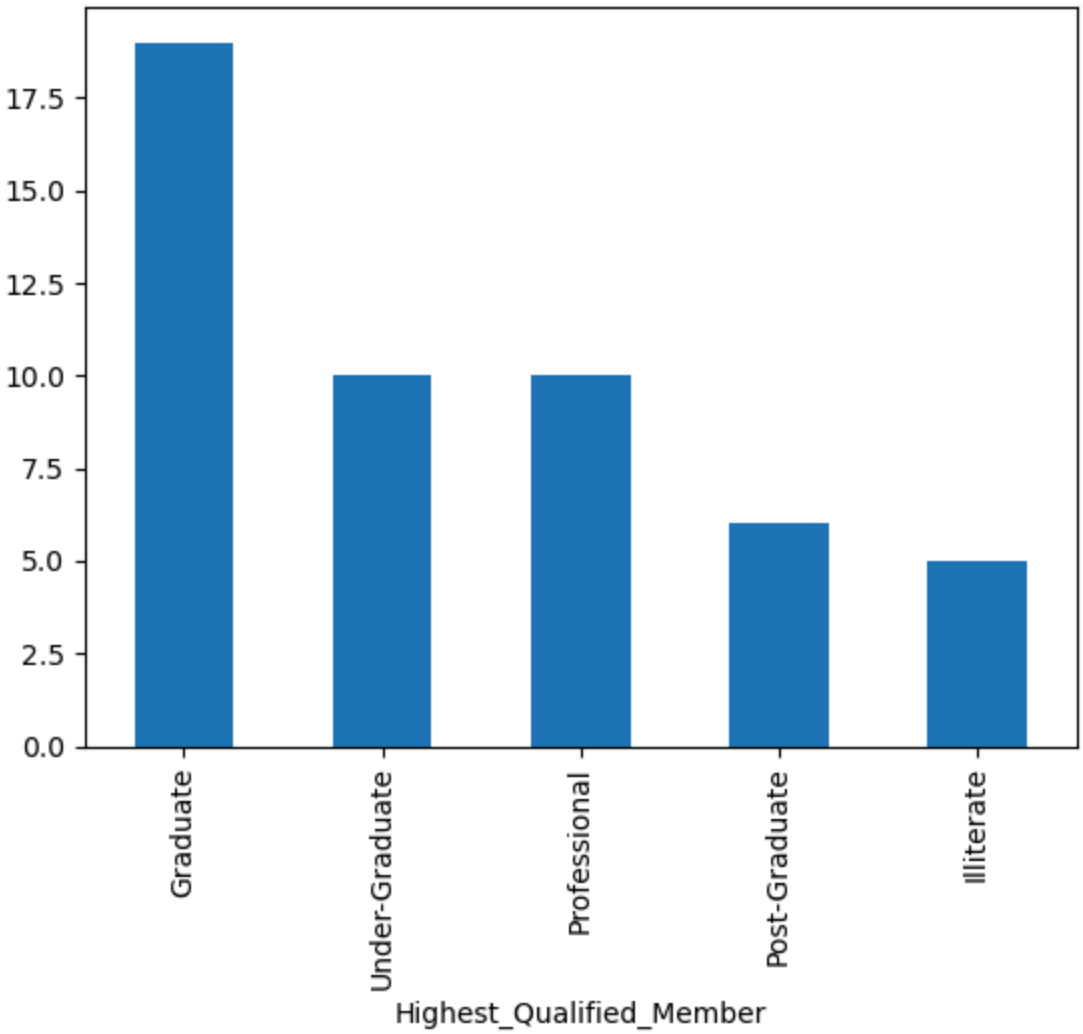
Out[12]:

col_0	Mthly_HH_Expense	count
18	25000	8

Plot the Histogram to Count the Highest qualified member

In [15]:
```
income_df["Highest_Qualified_Member"].value_counts().plot(kind="bar")
```

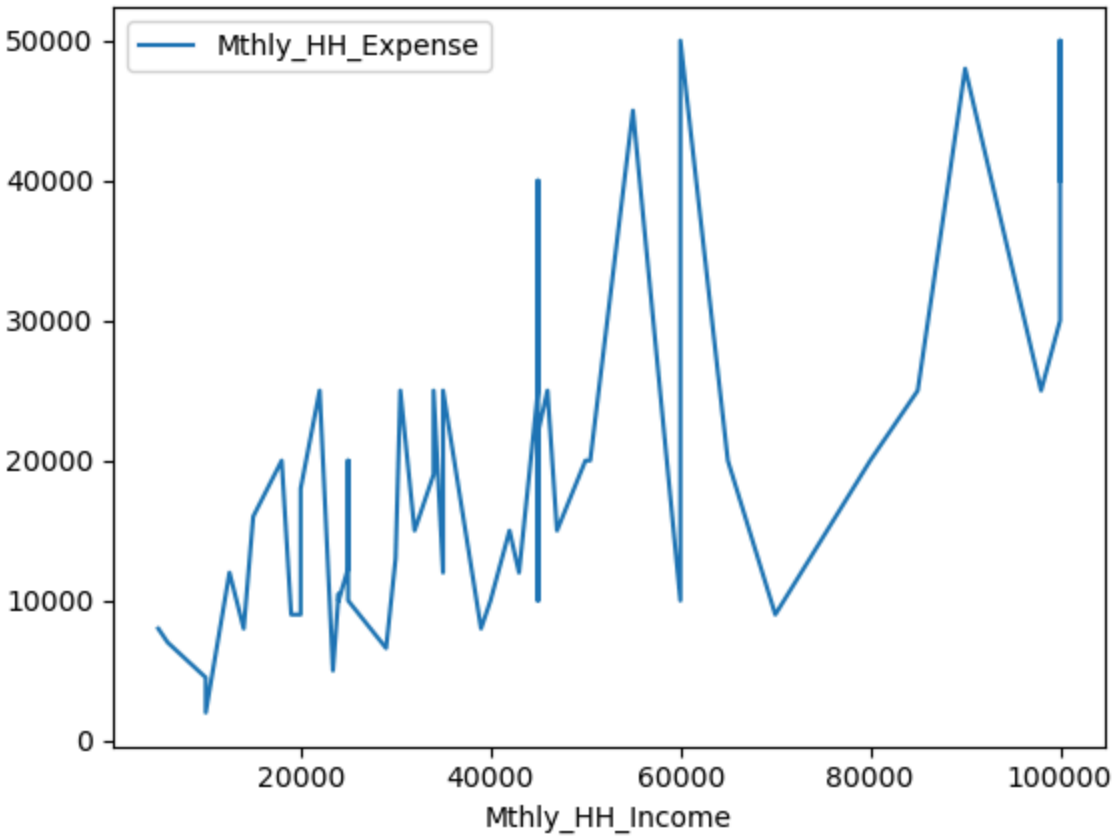
Out[15]: <Axes: xlabel='Highest_Qualified_Member'>



Calculate IQR (difference between 75% and 25% quartile)

In [17]:
```
income_df.plot(x="Mthly_HH_Income", y="Mthly_HH_Expense")
IQR = income_df["Mthly_HH_Expense"].quantile(0.75)-income_df["Mthly_HH_Expense"].qu
IQR
```

Out[17]: 15000.0



Calculate Standard Deviation for first 4 Columns

In [19]:
```
pd.Dataframe(income_df.iloc[:0:4].var().to_frame())
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[19], line 1
----> 1 pd.Dataframe(income_df.iloc[:0:4].var().to_frame())

AttributeError: module 'pandas' has no attribute 'Dataframe'
```

In [ ]:
```
```