In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [4]:
```
df=pd.read_csv(r"C:\Users\HP\Downloads\Inc_Exp_Data.csv")
```

In [5]:
```
df
```

Out[5]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annu
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	
30	45000	25000	6	0	
31	45000	40000	6	3500	
32	45000	10000	2	1000	

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annu
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 [6]:
```
df.head()
```

Out[6]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annua
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 [7]:
```
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 [9]:
```
df.shape
```

Out[9]: (50, 7)

In [10]:
```
df.describe()
```

Out[10]:

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

In [11]:
```
df.transpose()
```

Out[11]:

	0	1	2	3	4	5	
Mthly_HH_Income	5000	6000	10000	10000	12500	14000	
Mthly_HH_Expense	8000	7000	4500	2000	12000	8000	
No_of_Fly_Members	3	2	2	1	2	2	
Emi_or_Rent_Amt	2000	3000	0	0	3000	0	
Annual_HH_Income	64200	79920	112800	97200	147000	196560	
Highest_Qualified_Member	Under-Graduate	Illiterate	Under-Graduate	Illiterate	Graduate	Graduate	G
No_of_Earning_Members	1	1	1	1	1	1	

7 rows × 50 columns

In [12]:
```
df.isna().any()
```

Out[12]:
```
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
```

In [13]:
```
df['Mthly_HH_Expense'].median()
```

Out[13]:
```
np.float64(15500.0)
```

In [15]:
```
df['Mthly_HH_Expense'].mean()
```

Out[15]:
```
np.float64(18818.0)
```

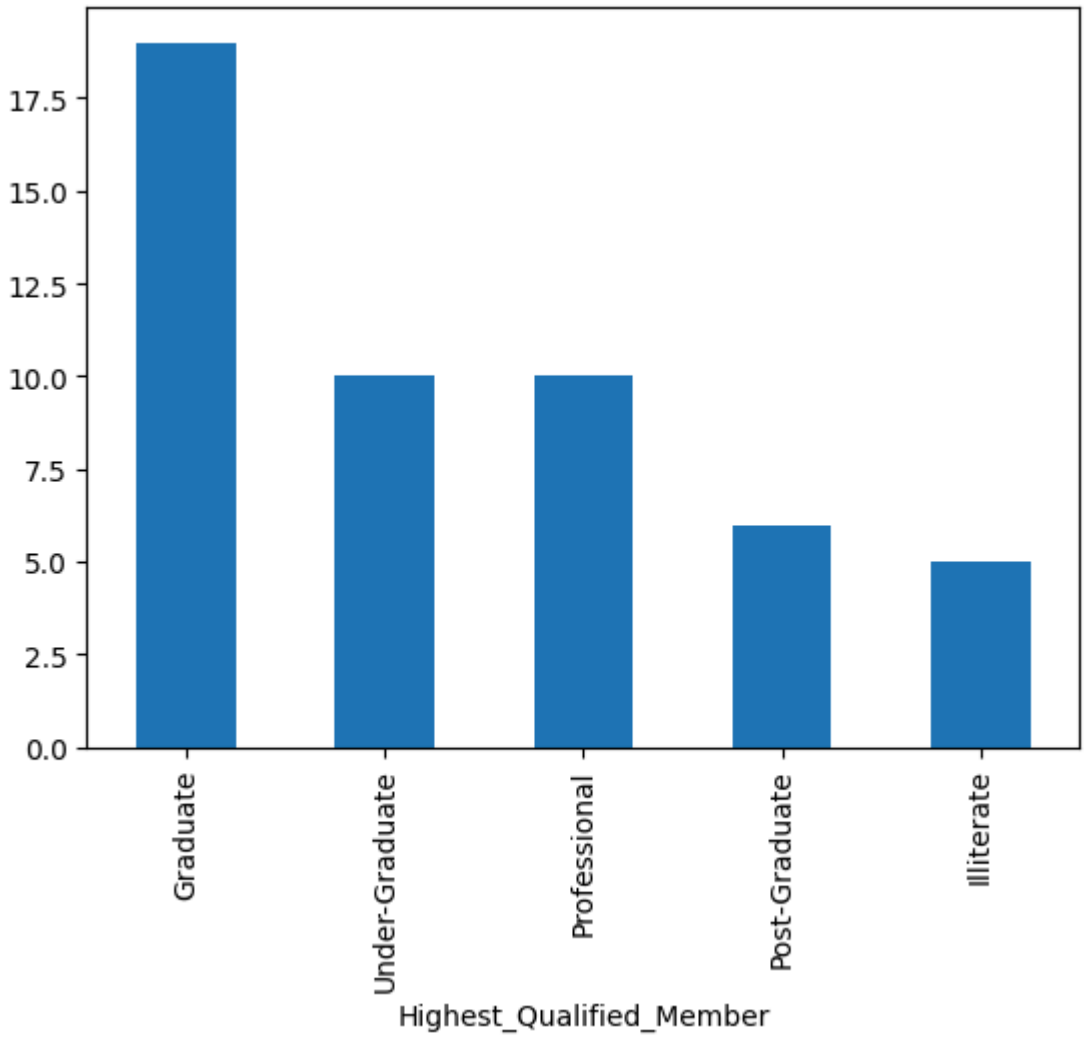
In [25]:
```
mth_exp_tmp=pd.crosstab(index=df['Mthly_HH_Expense'],columns='count')
mth_exp_tmp.reset_index(inplace=True)
mth_exp_tmp[mth_exp_tmp['count']==df.Mthly_HH_Expense.value_counts().max()]
```

Out[25]:

col_0	Mthly_HH_Expense	count
18	25000	8

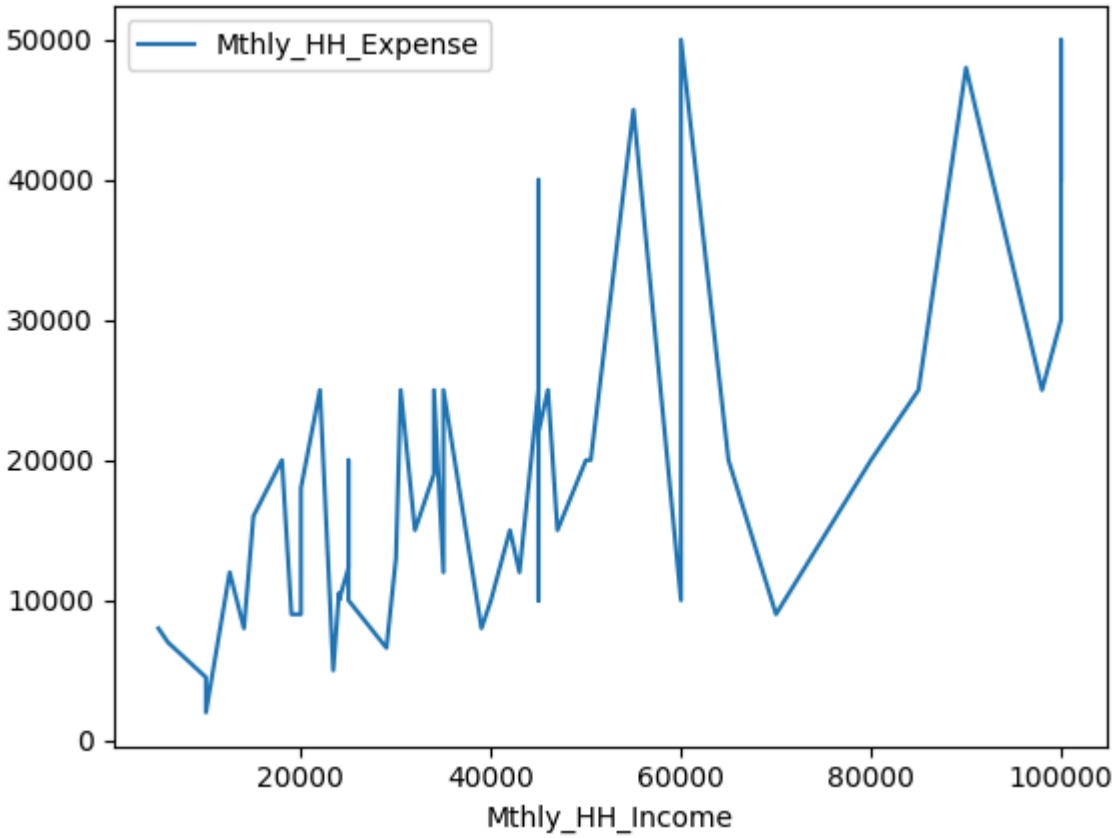
In [24]:
```
df['Highest_Qualified_Member'].value_counts().plot(kind='bar')
```

Out[24]:
```
<Axes: xlabel='Highest_Qualified_Member'>
```



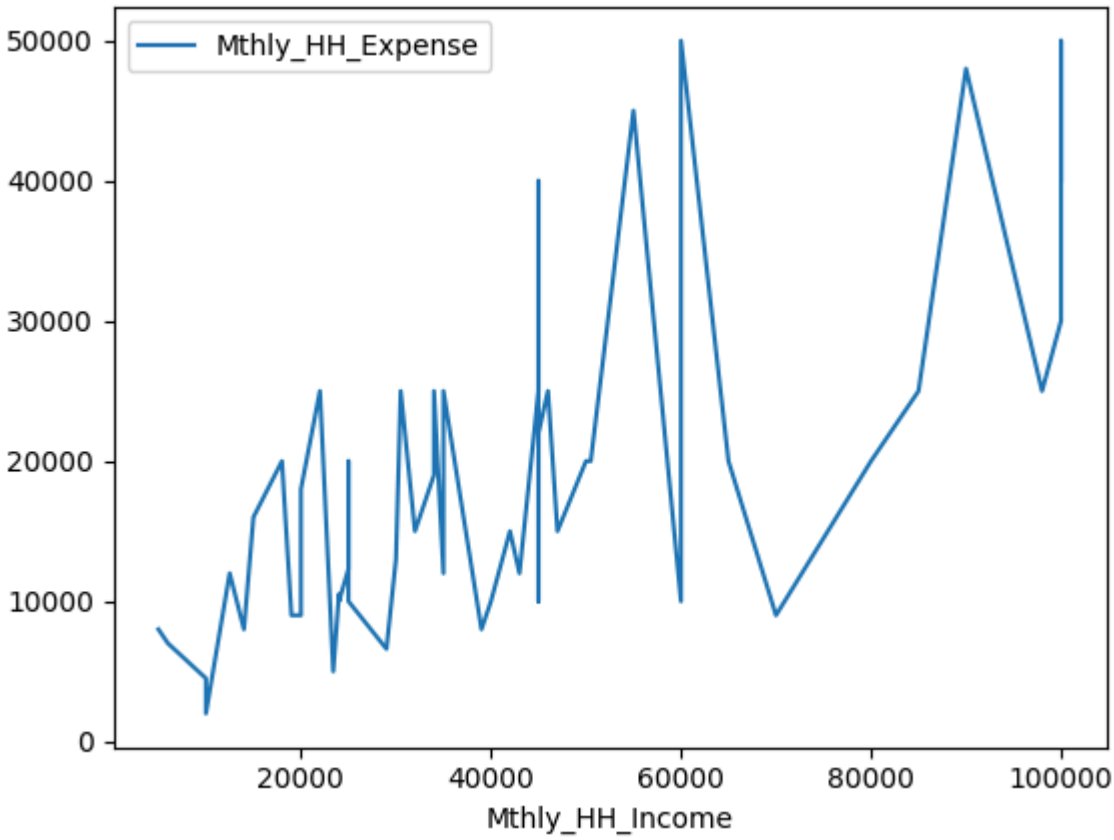
In [26]:
```
df.plot(x='Mthly_HH_Income',y='Mthly_HH_Expense')
IQR=df['Mthly_HH_Expense'].quantile(0.75)-df['Mthly_HH_Expense'].quantile(0.25)
IQR
```

Out[26]:
```
np.float64(15000.0)
```



In [27]:
```
df.plot(x='Mthly_HH_Income',y='Mthly_HH_Expense')
IQR=df['Mthly_HH_Expense'].quantile(0.50)-df['Mthly_HH_Expense'].quantile(0.25)
IQR
```

Out[27]: np.float64(5500.0)



In [28]:
```
pd.DataFrame(df.iloc[:,0:5].std().to_frame())
```

Out[28]:

	0
Mthly_HH_Income	26097.908979
Mthly_HH_Expense	12090.216824
No_of_Fly_Members	1.517382
Emi_or_Rent_Amt	6241.434948
Annual_HH_Income	320135.792123

In [29]:
```
pd.DataFrame(df.iloc[:,0:5].std().to_frame()).T
```

Out[29]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt	Annua
0	26097.908979	12090.216824	1.517382	6241.434948	3

In [30]:
```
pd.DataFrame(df.iloc[:,0:4].var().to_frame()).T
```

Out[30]:

	Mthly_HH_Income	Mthly_HH_Expense	No_of_Fly_Members	Emi_or_Rent_Amt
0	6.811009e+08	1.461733e+08	2.302449	3.895551e+07

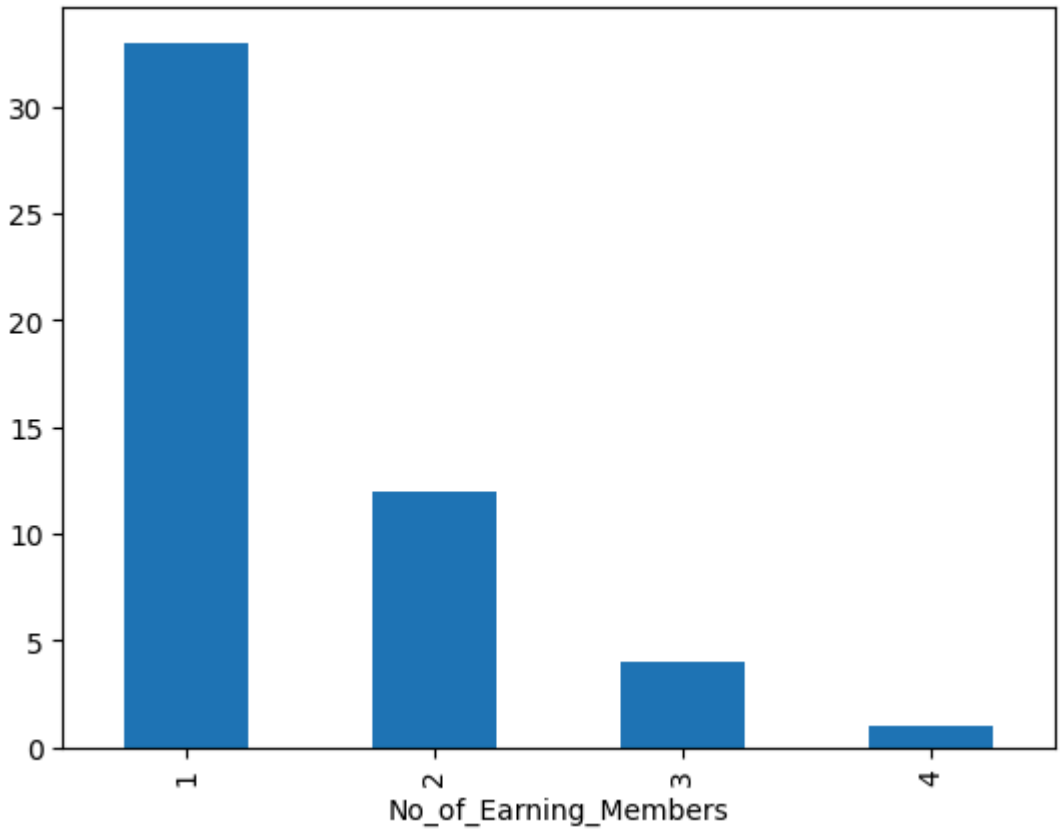
In [32]:
```
df['Highest_Qualified_Member'].value_counts().to_frame().T
```

Out[32]:

Highest_Qualified_Member	Graduate	Under-Graduate	Professional	Post-Graduate	Illiterate
count	19	10	10	6	5

In [33]:
```
df['No_of_Earning_Members'].value_counts().plot(kind='bar')
```

Out[33]:
```
<Axes: xlabel='No_of_Earning_Members'>
```



In [ ]: