

```
In [14]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
In [15]: df=pd.read_csv('Downloads/Housing3/Transformed_Housing_Data.csv')
```

```
In [16]: df.head()
```

Out[16]:

	ID	Sale Price	No of Bedrooms	No of Bathrooms	Flat Area (in Sqft)	Lot Area (in Sqft)	No of Floors	Waterfront View	No of Times Visited	Condition of the House	...	Area of the House from Basement (in Sqft)	Basement Area (in Sqft)	Age of House (in Years)	Zipcode	Latitude	Longitude	Living Area after Renovation (in Sqft)	Lot Area after Renovation (in Sqft)	Ever Renovated	Years Since Renovation
0	7129300520	221900.0	3	1.00	1180.0	5650.0	1.0	No	0	Fair	...	1180.0	0	63	98178.0	47.5112	-122.257	1340.0	5650	No	0
1	6414100192	538000.0	3	2.25	2570.0	7242.0	2.0	No	0	Fair	...	2170.0	400	67	98125.0	47.7210	-122.319	1690.0	7639	Yes	26
2	5631500400	180000.0	2	1.00	770.0	10000.0	1.0	No	0	Fair	...	770.0	0	85	98028.0	47.7379	-122.233	2720.0	8062	No	0
3	2487200875	604000.0	4	3.00	1960.0	5000.0	1.0	No	0	Excellent	...	1050.0	910	53	98136.0	47.5208	-122.393	1360.0	5000	No	0
4	1954400510	510000.0	3	2.00	1680.0	8080.0	1.0	No	0	Fair	...	1680.0	0	31	98074.0	47.6168	-122.045	1800.0	7503	No	0

5 rows × 21 columns

```
In [5]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21699 entries, 0 to 21698
Data columns (total 21 columns):
#   Column                Non-Null Count  Dtype
---  -
0   ID                    21699 non-null  int64
1   Sale Price            21699 non-null  float64
2   No of Bedrooms        21699 non-null  int64
3   No of Bathrooms       21699 non-null  float64
4   Flat Area (in Sqft)    21699 non-null  float64
5   Lot Area (in Sqft)     21699 non-null  float64
6   No of Floors           21699 non-null  float64
7   Waterfront View        21699 non-null  object
8   No of Times Visited    21699 non-null  int64
9   Condition of the House 21699 non-null  object
10  Overall Grade          21699 non-null  int64
11  Area of the House from Basement (in Sqft) 21699 non-null  float64
12  Basement Area (in Sqft) 21699 non-null  int64
13  Age of House (in Years) 21699 non-null  int64
14  Zipcode                21699 non-null  float64
15  Latitude                21699 non-null  float64
16  Longitude                21699 non-null  float64
17  Living Area after Renovation (in Sqft) 21699 non-null  float64
18  Lot Area after Renovation (in Sqft) 21699 non-null  int64
19  Ever Renovated          21699 non-null  object
20  Years Since Renovation 21699 non-null  int64
dtypes: float64(18), int64(8), object(3)
memory usage: 3.5+ MB
```

```
In [6]: df.drop(columns=['ID'],inplace=True)
```

```
In [8]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21699 entries, 0 to 21698
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sale Price            21699 non-null  float64
1   No of Bedrooms        21699 non-null  int64
2   No of Bathrooms       21699 non-null  float64
3   Flat Area (in Sqft)    21699 non-null  float64
4   Lot Area (in Sqft)     21699 non-null  float64
5   No of Floors           21699 non-null  float64
6   Waterfront View        21699 non-null  object
7   No of Times Visited    21699 non-null  int64
8   Condition of the House 21699 non-null  object
9   Overall Grade          21699 non-null  int64
10  Area of the House from Basement (in Sqft) 21699 non-null  float64
11  Basement Area (in Sqft) 21699 non-null  int64
12  Age of House (in Years) 21699 non-null  int64
13  Zipcode                21699 non-null  float64
14  Latitude                21699 non-null  float64
15  Longitude                21699 non-null  float64
16  Living Area after Renovation (in Sqft) 21699 non-null  float64
17  Lot Area after Renovation (in Sqft) 21699 non-null  int64
18  Ever Renovated          21699 non-null  object
19  Years Since Renovation 21699 non-null  int64
dtypes: float64(18), int64(7), object(3)
memory usage: 3.3+ MB
```

```
In [8]: df['Condition of the House'].head()
```

```
Out[8]: 0      Fair
1      Fair
2      Fair
3    Excellent
4      Fair
Name: Condition of the House, dtype: object
```

```
In [9]: df['Condition of the House'].unique()
```

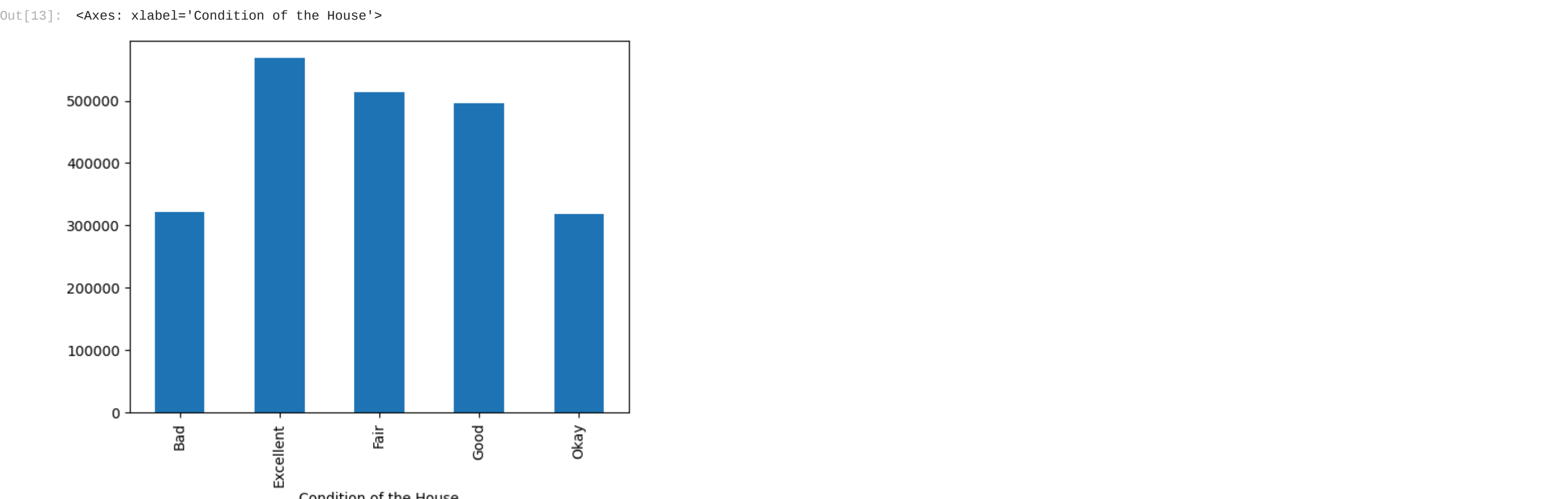
```
Out[9]: array(['Fair', 'Excellent', 'Good', 'Bad', 'Okay'], dtype=object)
```

```
In [12]: df['Condition of the House'].value_counts()
```

```
Out[12]: Condition of the House
Fair      14928
Good      5678
Excellent 1791
Okay     172
Bad        30
Name: count, dtype: int64
```

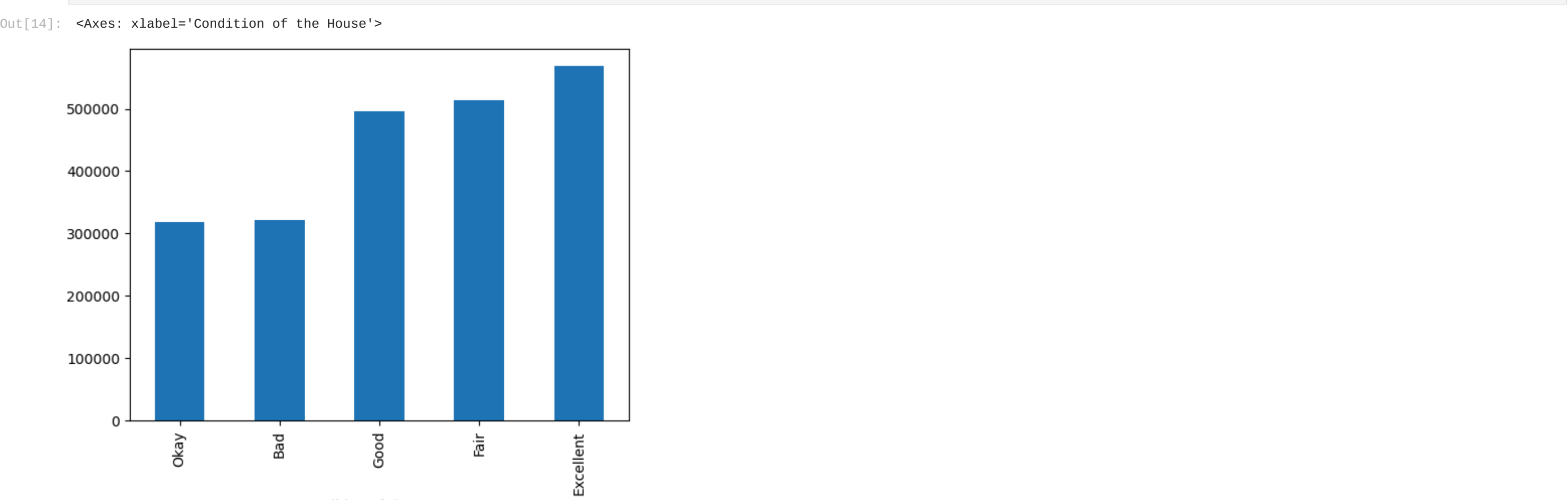
```
In [13]: df.groupby('Condition of the House')['Sale Price'].mean().plot(kind='bar')
```

```
Out[13]: <Axes: xlabel='Condition of the House'>
```



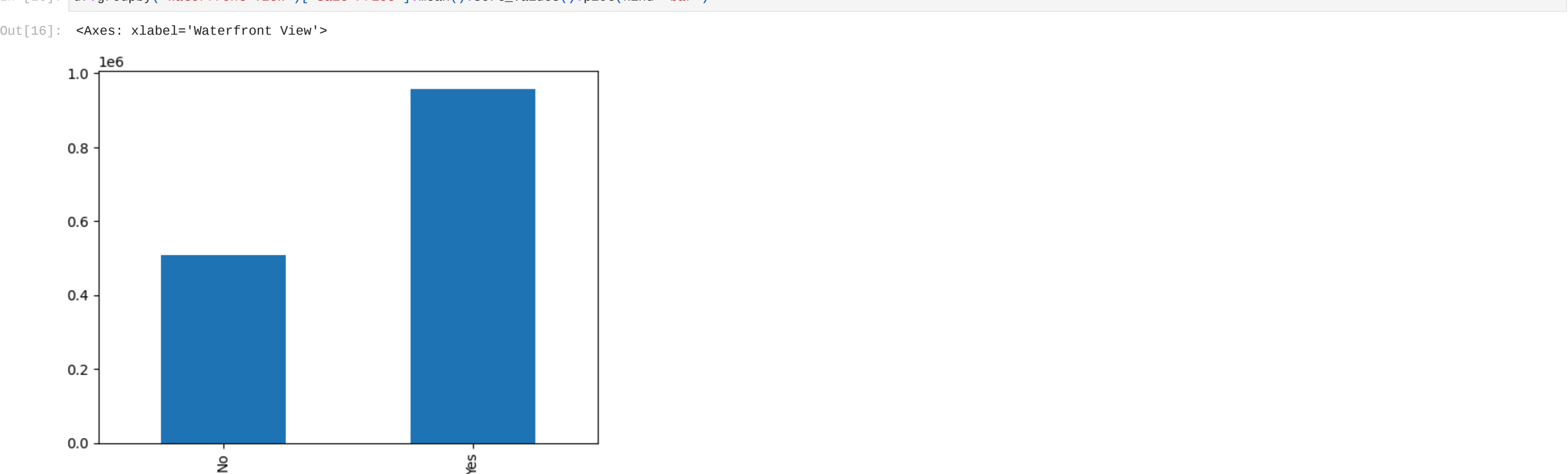
```
In [14]: df.groupby('Condition of the House')['Sale Price'].mean().sort_values().plot(kind='bar')
```

```
Out[14]: <Axes: xlabel='Condition of the House'>
```



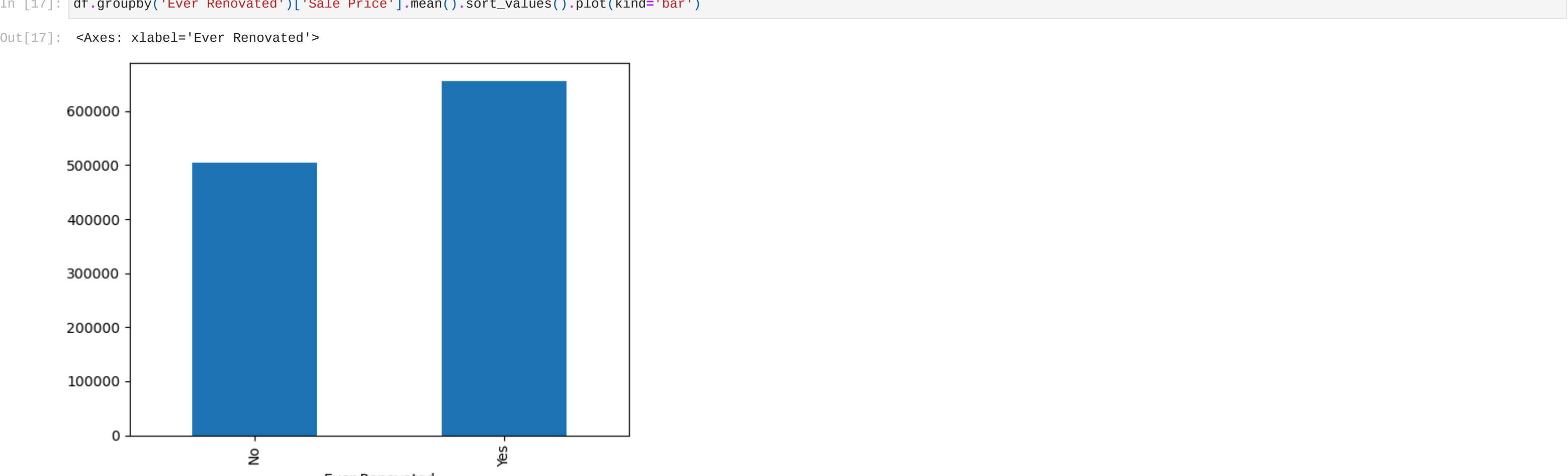
```
In [16]: df.groupby('Waterfront View')['Sale Price'].mean().sort_values().plot(kind='bar')
```

```
Out[16]: <Axes: xlabel='Waterfront View'>
```



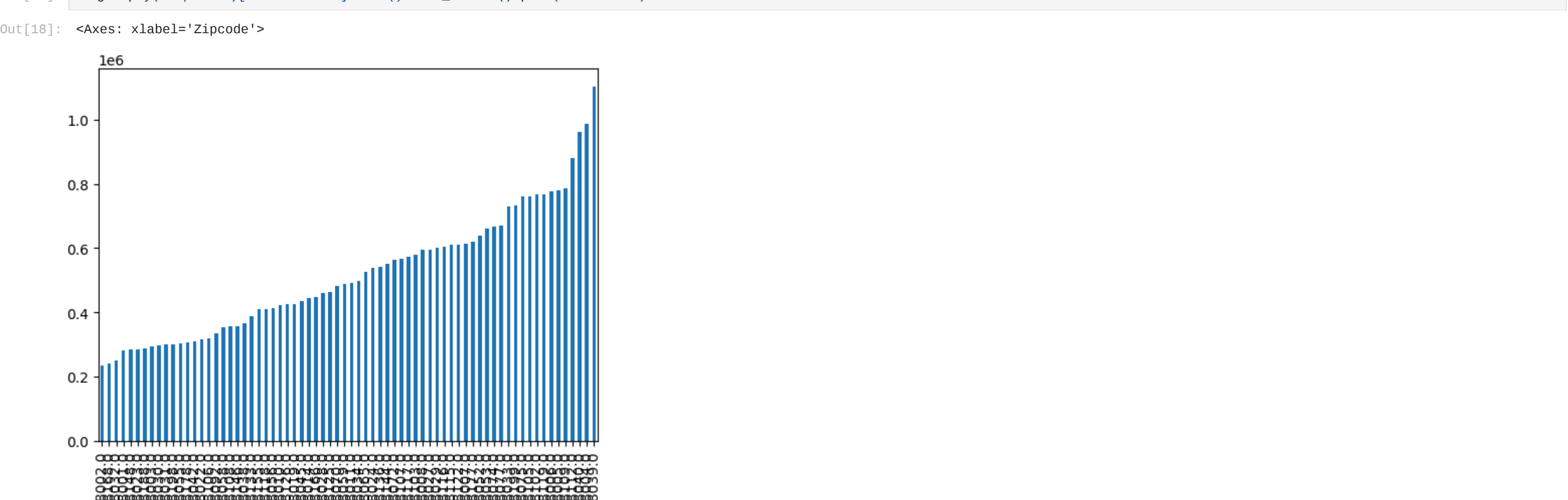
```
In [17]: df.groupby('Ever Renovated')['Sale Price'].mean().sort_values().plot(kind='bar')
```

```
Out[17]: <Axes: xlabel='Ever Renovated'>
```



```
In [18]: df.groupby('Zipcode')['Sale Price'].mean().sort_values().plot(kind='bar')
```

```
Out[18]: <Axes: xlabel='Zipcode'>
```



```
In [17]: from statsmodels.formula.api import ols
import statsmodels.api as sm
```

```
In [29]: df=df.rename(columns={'Condition of the House':'Condition_of_the_House'})
```

```
In [38]: df=df.rename(columns={'Ever Renovated':'Ever_Renovated'})
```

```
In [31]: df=df.rename(columns={'Waterfront View':'Waterfront_View'})
```

```
In [32]: df=df.rename(columns={'Sale Price':'Sale_Price'})
```

```
In [33]: df.head()
```

Out[33]:

	ID	Sale_Price	No of Bedrooms	No of Bathrooms	Flat Area (in Sqft)	Lot Area (in Sqft)	No of Floors	Waterfront_View	No of Times Visited	Condition_of_the_House	...	Area of the House from Basement (in Sqft)	Basement Area (in Sqft)	Age of House (in Years)	Zipcode	Latitude	Longitude	Living Area after Renovation (in Sqft)	Lot Area after Renovation (in Sqft)	Ever_Renovated
0	7129300520	221900.0	3	1.00	1180.0	5650.0	1.0	No	0	Fair	...	1180.0	0	63	98178.0	47.5112	-122.257	1340.0	5650	No
1	6414100192	538000.0	3	2.25	2570.0	7242.0	2.0	No	0	Fair	...	2170.0	400	67	98125.0	47.7210	-122.319	1690.0	7639	Yes
2	5631500400	180000.0	2	1.00	770.0	10000.0	1.0	No	0	Fair	...	770.0	0	85	98028.0	47.7379	-122.233	2720.0	8062	No
3	2487200875	604000.0	4	3.00	1960.0	5000.0	1.0	No	0	Excellent	...	1050.0	910	53	98136.0	47.5208	-122.393	1360.0	5000	No
4	1954400510	510000.0	3	2.00	1680.0	8080.0	1.0	No	0	Fair	...	1680.0	0	31	98074.0	47.6168	-122.045	1800.0	7503	No

5 rows × 21 columns

```
In [22]: df['Waterfront_View'].unique()
```

```
Out[22]: array(['No', 'Yes'], dtype=object)
```

```
In [34]: mod=ols('Sale_Price ~ Condition_of_the_House',data=df).fit()
```

```
In [35]: Anova_Table=sm.stats.anova_lm(mod,tyo=2)
```

```
In [36]: print(Anova_Table)
```

	df	sum_sq	mean_sq	F	\
Condition_of_the_House	4.0	1.441831e+13	3.604579e+12	58.25563	
Residual	21694.0	1.336752e+15	6.187520e+10	NaN	
					PR(>F)
Condition_of_the_House	5.447695e-49				
Residual	NaN				

```
In [38]: df=pd.get_dummies(df,columns=['Condition_of_the_House'],drop_first=True)
```

```
In [39]: df.head()
```

Out[39]:

	ID	Sale_Price	No of Bedrooms	No of Bathrooms	Flat Area (in Sqft)	Lot Area (in Sqft)	No of Floors	Waterfront_View	No of Times Visited	Overall Grade	...	Latitude	Longitude	Living Area after Renovation (in Sqft)	Lot Area after Renovation (in Sqft)	Ever_Renovated	Years Since Renovation	Condition_of_the_House_Excellent	Condition_of_the_House_Okay
0	7129300520	221900.0	3	1.00	1180.0	5650.0	1.0	No	0	7	...	47.5112	-122.257	1340.0	5650	No	0	False	True
1	6414100192	538000.0	3	2.25	2570.0	7242.0	2.0	No	0	7	...	47.7210	-122.319	1690.0	7639	Yes	26	True	False
2	5631500400	180000.0	2	1.00	770.0	10000.0	1.0	No	0	6	...	47.7379	-122.233	2720.0	8062	No	0	False	True
3	2487200875	604000.0	4	3.00	1960.0	5000.0	1.0	No	0	7	...	47.5208	-122.393	1360.0	5000	No	0	True	False
4	1954400510	510000.0	3	2.00	1680.0	8080.0	1.0	No	0	8	...	47.6168	-122.045	1800.0	7503	No	0	False	True

5 rows × 24 columns

