Flipkart Air Conditioners Analysis

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

This Dataset contains tabular data of all air conditioners listed on Flipkart.Details included are price details, ratings and features.All the product details is scraped from the Flipkart's official website.

In [2]:
```
df = pd.read_csv("Air Conditioners.csv")
df.head()
```

Out[2]:

	name	main_category	sub_category	image	link	ratings	no_of_ratings	discount_price	actual_price
0	Lloyd 1.5 Ton 3 Star Inverter Split Ac (5 In 1...	appliances	Air Conditioners	https://m.media-amazon.com/images/I/31UISB90sY...	https://www.amazon.in/Lloyd-Inverter-Convertib...	4.2	2,255	₹32,999	₹58,990
1	LG 1.5 Ton 5 Star AI DUAL Inverter Split AC (C...	appliances	Air Conditioners	https://m.media-amazon.com/images/I/51JFb7FctD...	https://www.amazon.in/LG-Convertible-Anti-Viru...	4.2	2,948	₹46,490	₹75,990
2	LG 1 Ton 4 Star Ai Dual	appliances	Air Conditioners	https://m.media-amazon.com/images/I/51JFb7FctD	https://www.amazon.in/LG-Inverter-Convertible-	4.2	1,206	₹34,490	₹61,990

[3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 720 entries, 0 to 719
Data columns (total 9 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   name            720 non-null    object
 1   main_category   720 non-null    object
 2   sub_category    720 non-null    object
 3   image           720 non-null    object
 4   link            720 non-null    object
 5   ratings         433 non-null    object
 6   no_of_ratings   433 non-null    object
 7   discount_price  457 non-null    object
 8   actual_price    500 non-null    object
dtypes: object(9)
memory usage: 50.8+ KB
```

Understanding Given Data -

1. name - These are Names of air conditioners
2. main_category - These are category of products
3. sub_category - These are sub categories of products
4. image - These column has images of products
5. link - These column has link of products
6. ratings - These column stores ratings of products
7. no_of_ratings - These column has no of ratings given to a particular product
8. discount_price - Discount of every product
9. actual_price - This column shows actual price of product

Cleaning Data

Droping Columns image and link as we don't need this columns in Analysis

[4]:
```
df.drop(['image','link'],axis=1,inplace=True)
```

Here we have 3 values that we have to change as we need to convert these columns into numeric, so we will replace them with null values

Replacing 'get', 'Only 2 left in stock.','Only 1 left in stock.', with null values

[5]:
```
df.replace('Get',np.nan,inplace=True)
df.replace('Only 2 left in stock.',np.nan,inplace=True)
df.replace('Only 1 left in stock.',np.nan,inplace=True)
```

Finding null/missing values in the data if any.

[6]:
```
df.isnull().sum()/len(df)*100
```

[6]:
```
name               0.000000
main_category      0.000000
sub_category       0.000000
ratings           41.250000
no_of_ratings     41.250000
discount_price    36.527778
actual_price      30.555556
dtype: float64
```

As we can see there are null values present in 4 columns.

before handling null values we will Replace ' , ' , '₹' with nothing.

[7]:
```
df['no_of_ratings'] = df['no_of_ratings'].str.replace(',','')

df['discount_price'] = df['discount_price'].str.replace('₹','')
df['discount_price'] = df['discount_price'].str.replace(',','')

df['actual_price'] = df['actual_price'].str.replace('₹','')
df['actual_price'] = df['actual_price'].str.replace(',','')
```

Changing Datatypes of columns 'no_of_ratings','ratings','discount_price','actual_price' from object to float

[8]:
```
df['no_of_ratings'] = df['no_of_ratings'].astype(float)

df['ratings'] = df['ratings'].astype(float)

df['discount_price'] = df['discount_price'].astype(float)

df['actual_price'] = df['actual_price'].astype(float)
```

Filling Missing Values:

[9]:
```
df.fillna(df.mean(),inplace=True)
```

10]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 720 entries, 0 to 719
Data columns (total 7 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   name            720 non-null    object 
 1   main_category   720 non-null    object 
 2   sub_category    720 non-null    object 
 3   ratings         720 non-null    float64
 4   no_of_ratings   720 non-null    float64
 5   discount_price  720 non-null    float64
 6   actual_price    720 non-null    float64
dtypes: float64(4), object(3)
memory usage: 39.5+ KB
```

As We can this Dataset is Cleaned and now we can do Analysis on this Dataset

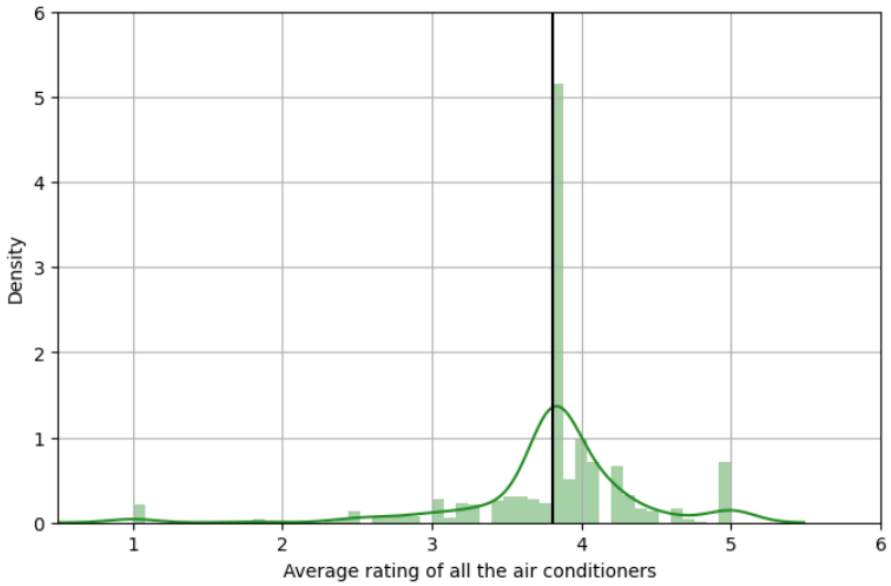
Analysis

1) What is the average rating of all the air conditioners?

In [12]:
```
average_rating = df['ratings'].mean()
print("Average rating of all the air conditioners:", average_rating)
```

```
Average rating of all the air conditioners: 3.809692671394799
```

In [13]:
```
lm=df['ratings'].mean()
plt.figure(figsize=(8,5))
sns.distplot(df['ratings'],color='forestgreen')
plt.axvline(lm,color='black')
plt.xlabel('Average rating of all the air conditioners')
plt.xlim((0.5,6))
plt.ylim(0,6)
plt.grid()
plt.show()
```



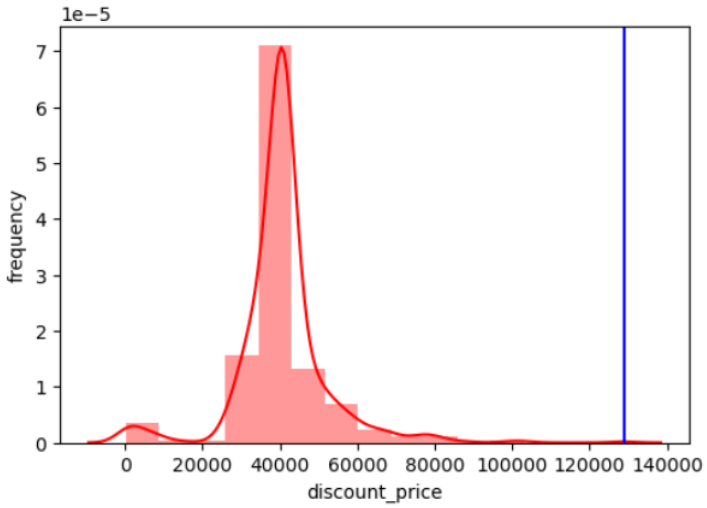
The graph above shows us the average of ratings of all air conditioners wherein maximum rating is 5.0, minimum rating is 1.0 and average rating is 3.809

2) What is the highest discount price among all the air conditioners?

[14]:
```
max_discount_price = df['discount_price'].max()
print("Highest discount price among all the air conditioners:", max_discount_price)
```

```
Highest discount price among all the air conditioners: 128800.0
```

[15]:
```
lmax=df['discount_price'].max()
plt.figure(figsize=(6,4))
sns.distplot(df['discount_price'],color='red',bins=15)
plt.axvline(lmax,color='blue')
plt.xlabel('discount_price')
plt.ylabel('frequency')
plt.show()
```



The graph above shows us the discount price of air conditioners wherein maximum discount price is 128800.0 and minimum discount price is 199.0

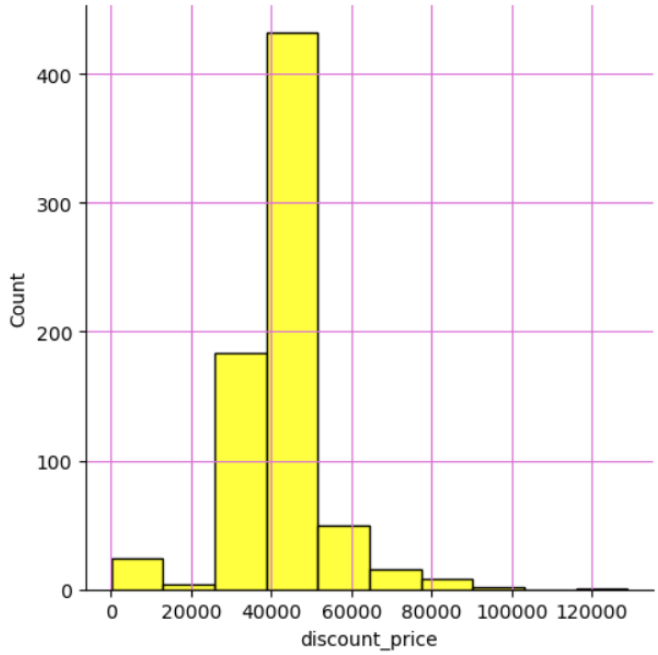
3) How many air conditioners have a discount price greater than 40000?

[16]:
```
AC = len(df[df['discount_price'] > 40000])
print("Number of air conditioners with a discount price greater than 40000:", AC)
```

```
Number of air conditioners with a discount price greater than 40000: 464
```

[17]:
```
plt.figure(figsize=(5,7))
sns.displot(df['discount_price'],color='yellow',bins=10)
plt.grid(color='orchid')
plt.show()
```

```
<Figure size 500x700 with 0 Axes>
```



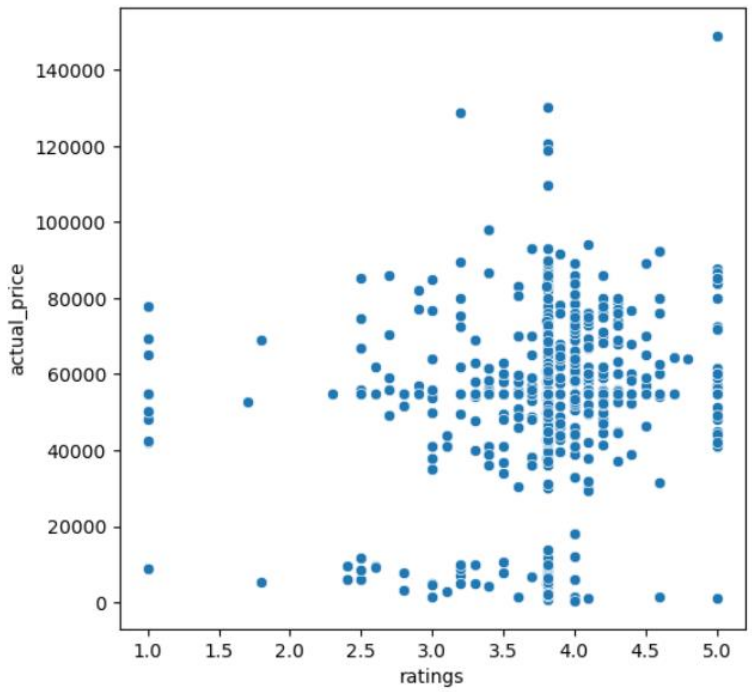
This graph shows us the discount price of air conditioners and their frequencies wherein discount price of 464 air conditioners is more than 40000

4) Display the Actual price of the AC where the Rating less than 3

8]:
```
df['actual_price'].loc[df['ratings']<=3]
```

9]:
```
plt.figure(figsize=(6,6))
sns.scatterplot(data=df,x='ratings',y='actual_price',marker='o')
```

9]:
```
<Axes: xlabel='ratings', ylabel='actual_price'>
```



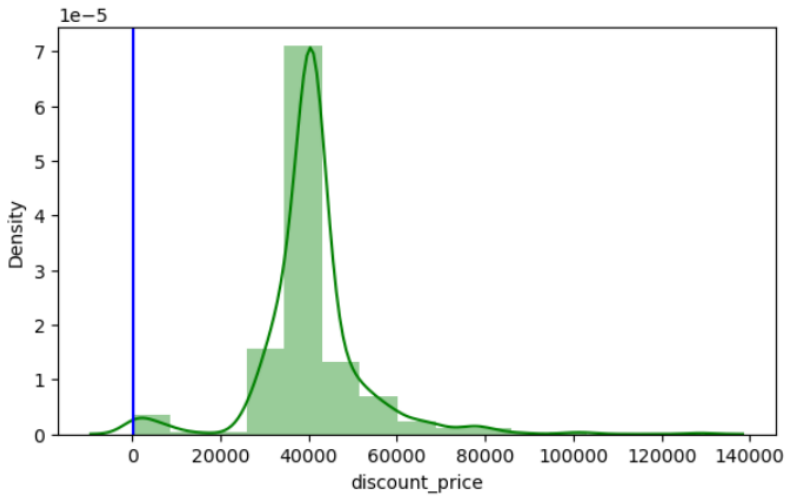
This graph shows actual price of air conditioners vs ratings wherein highest actual price of 3 star rated air conditioners is above 140000 and lowest is around 1300

5) What is the lowest discount price among all the air conditioners?

[20]:
```
min_discount_price = df['discount_price'].min()
print("Lowest discount price among all the air conditioners:", min_discount_price)
```

```
Lowest discount price among all the air conditioners: 199.0
```

[21]:
```
lmin=df['discount_price'].min()
plt.figure(figsize=(7,4))
sns.distplot(df['discount_price'],color='green',bins=15)
plt.axvline(lmin,color='blue')
plt.xlabel('discount_price')
plt.show()
```



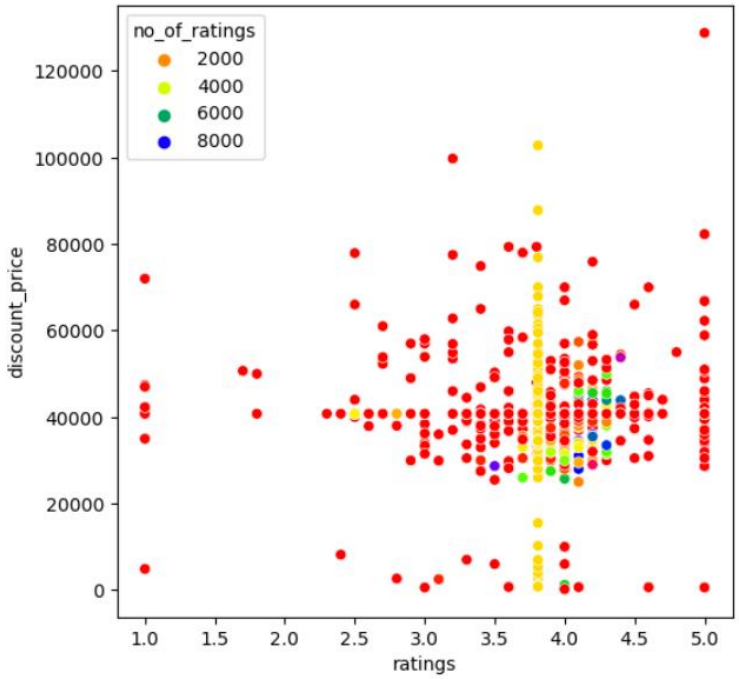
The graph above shows us the discount price of air conditioners wherein minimum discount price is 199.0 and maximum discount price is 128800.0

6) Display the Discount Price where Rating is Maximum

In [22]:
```
df['discount_price'].loc[df['ratings'].max()]
```

Out[22]:
```
31990.0
```

In [23]:
```
plt.figure(figsize=(6,6))
sns.scatterplot(data=df,x='ratings',y = 'discount_price',hue = 'no_of_ratings',palette='prism')
```



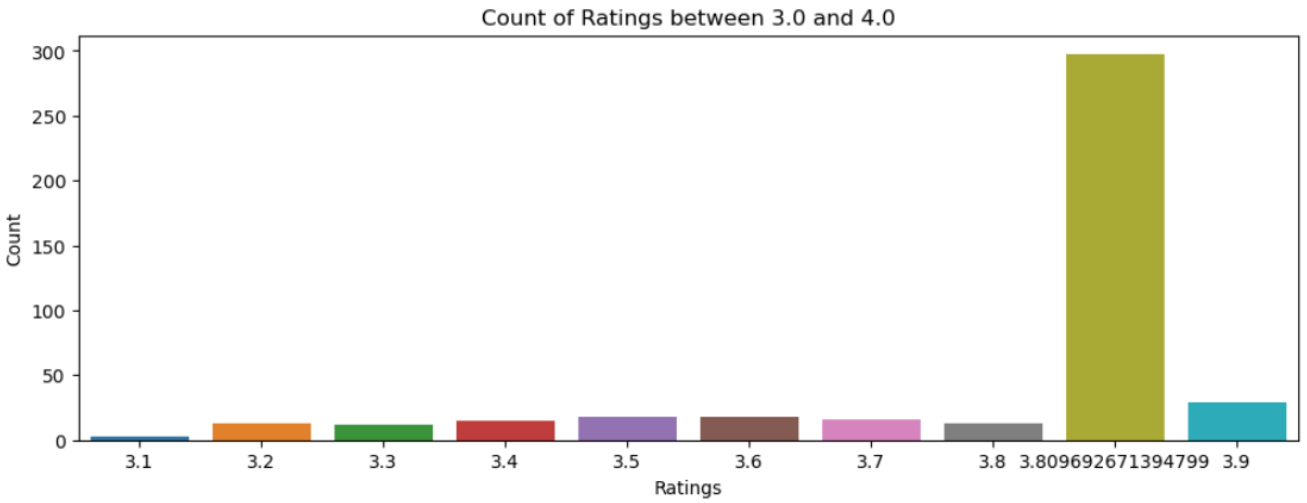
This graph shows discount price of air conditioners vs ratings wherein highest rating is 5.0 highest actual price of air conditioners is above 31990.0

7) How Many Air Conditioners have Ratings between 3 & 4

[24]:
```
df['ratings'].loc[(df['ratings']>3.0) & (df['ratings']<4.0)].value_counts()
```

t[24]:
```
3.809693    297
3.900000     29
3.500000     18
3.600000     18
3.700000     16
3.400000     15
3.800000     13
3.200000     13
3.300000     12
3.100000      3
Name: ratings, dtype: int64
```

[25]:
```
plt.figure(figsize=(12,4))
s = df.loc[(df['ratings'] > 3.0) & (df['ratings'] < 4.0)]
sns.countplot(data=s, x='ratings')
plt.xlabel('Ratings')
plt.ylabel('Count')
plt.title('Count of Ratings between 3.0 and 4.0')
plt.show()
```



This graph shows frequencies of Air Conditioners' ratings where maximum number of ratings are of 3.809693 and minimum are of 3.1

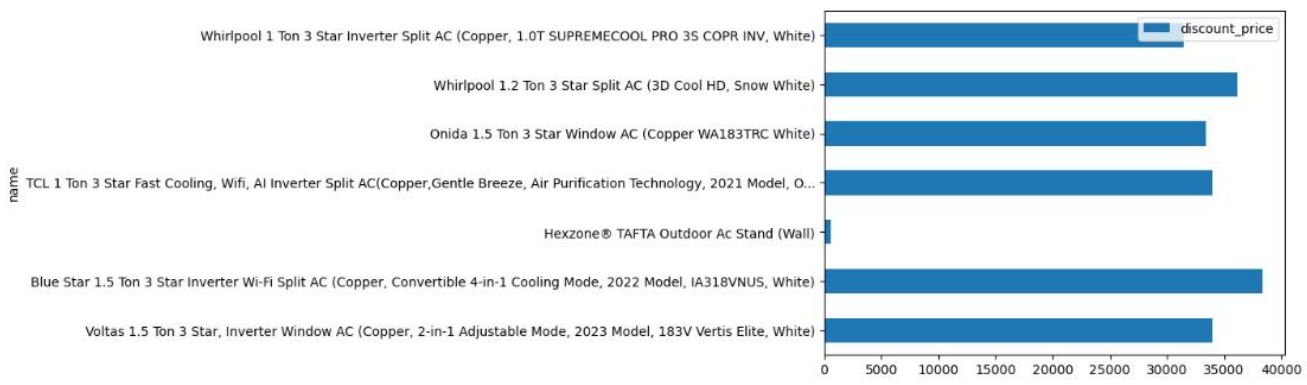
8) Display Names of AC's With Rating of 3 and having discount price less than 40000

[26]:
```
x = df[['name','discount_price']].loc[(df['ratings']==3.0) & (df['discount_price']<40000.0)]
x
```

t[26]:

	name	discount_price
78	Voltas 1.5 Ton 3 Star, Inverter Window AC (Cop...	33950.0
127	Blue Star 1.5 Ton 3 Star Inverter Wi-Fi Split ...	38390.0
243	Hexzone® TAFTA Outdoor Ac Stand (Wall)	599.0
404	TCL 1 Ton 3 Star Fast Cooling, Wifi, AI Invert...	33990.0
444	Onida 1.5 Ton 3 Star Window AC (Copper WA183TR...	33370.0
459	Whirlpool 1.2 Ton 3 Star Split AC (3D Cool HD,...	36189.0
670	Whirlpool 1 Ton 3 Star Inverter Split AC (Copp...	31490.0

[27]:
```
x.plot(x='name',y=['discount_price'],kind = 'barh')
```



This graph shows air conditioners' discount price having ratings of 3 and discount price below 40000 wherein minimum discountis of Hexagon AC and maximum is of Blue star AC

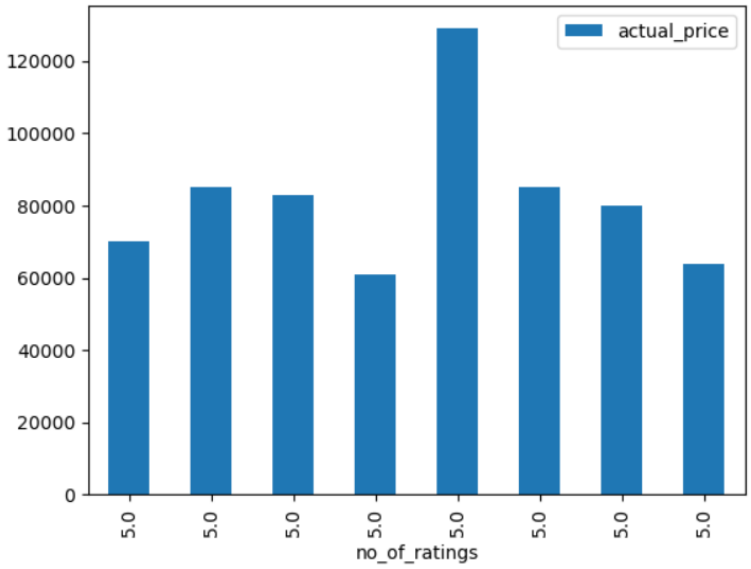
9) Display AC having price above 60,000 with Highest Rating

28]:
```
y = df.loc[(df['actual_price']>60000.0) & (df['no_of_ratings']==5.0)]
y
```

28]:

	name	main_category	sub_category	ratings	no_of_ratings	discount_price	actual_price
68	LG 1.5 Ton 3 Star DUAL Inverter Window AC (Cop...	appliances	Air Conditioners	4.2	5.0	37490.0	69990.0
113	Voltas 2 Ton 5 Star Inverter Split AC (Copper ...	appliances	Air Conditioners	5.0	5.0	58890.0	84990.0
155	MITSUBISHI HEAVY DUTY SRK25CSS-S6 2.2 Ton 3 St...	appliances	Air Conditioners	3.6	5.0	79350.0	83000.0
303	Lloyd 1.5 Ton 5 Star Split Inverter AC - White...	appliances	Air Conditioners	4.0	5.0	39499.0	60990.0
388	Voltas Venture Slimline Tower AC (3 Ton White)	appliances	Air Conditioners	3.2	5.0	99750.0	128900.0
392	Ogeneral 2 Ton 5 Star Split Inverter AC - Whit...	appliances	Air Conditioners	2.5	5.0	77890.0	85260.0
461	Voltas 24HY Hot and Cold Split AC (2 Ton 1 Sta...	appliances	Air Conditioners	3.2	5.0	56890.0	79990.0
587	Electrolux 1.5 Ton 3 Star Convertible Inverter...	appliances	Air Conditioners	4.0	5.0	38490.0	63990.0

29]:
```
y.plot(x='no_of_ratings',y=['actual_price'],kind = 'bar')
```



This graph show price vs ratings wherein all air conditioners have highest no of ratings and price of AC's are more than 60000 and maximum price is 128900.0 and minimum is 63990.0