

```
In [1]:
1 import numpy as np
2 import pandas as pd
3 from bs4 import BeautifulSoup as bs
4 import requests as re
5 import seaborn as sns
6 import matplotlib.pyplot as plt
```

```
In [2]:
1 url = 'https://www.flipkart.com/search?q=laptop&otracker=search&otracker=
```

```
In [3]:
1 response = re.get(url)
2 print(response)
```

<Response [200]>

```
In [4]:
1 flip = bs(response.text, 'html.parser')
2 print(flip)
```

```
<!DOCTYPE html>
<html lang="en"><head><link href="https://rukminim1.flixcart.com" rel="preconnect"/><link href="
-p-linchpin-web/fk-cp-zion/css/app_modules.chunk.905c37.css" rel="stylesheet"/><link href="//sta
nchpin-web/fk-cp-zion/css/app.chunk.e84a00.css" rel="stylesheet"/><meta content="text/html; char
e"/><meta content="IE=Edge" http-equiv="X-UA-Compatible"/><meta content="102988293558" property=
552,624500995,100000233612389" property="fb:admins"/><meta content="noodp" name="robots"/><link
528-140547-favicon-retina.ico" rel="shortcut icon"/><link href="/osdd.xml?v=2" rel="search" type
+xml"/><meta content="website" property="og:type"/><meta content="Flipkart.com" name="og_site_na
href="/apple-touch-icon-57x57.png" rel="apple-touch-icon" sizes="57x57"/><link href="/apple-touc
icon" sizes="72x72"/><link href="/apple-touch-icon-114x114.png" rel="apple-touch-icon" sizes="11
on-144x144.png" rel="apple-touch-icon" sizes="144x144"/><link href="/apple-touch-icon-57x57.png"
ent="app" name="twitter:card"/><meta content="@flipkart" name="twitter:site"/><meta content="@fl
ta content="Laptop- Buy Products Online at Best Price in India - All Categories | Flipkart.com"
="Shop for electronics, apparels & more using our Flipkart app Free shipping & COD." nam
ent="in" name="twitter:app:country"/><meta content="Flipkart" name="al:ios:app_name"/><meta cont
ore_id"/><meta content="Flipkart" name="twitter:app:name:iphone"/><meta content="742044692" name
tent="http://dl.flipkart.com/dl/home?" name="twitter:app:url:iphone"/><meta content="Flipkart" r
content="742044692" name="twitter:app:id:ipad"/><meta content="http://dl.flipkart.com/dl/home?"
content="Flipkart" name="twitter:app:name:googleplay"/><meta content="com.flipkart.android" name
content="http://dl.flipkart.com/dl/home?" name="twitter:app:url:googleplay"/><style>#container {
    height: 100%;
}</style> <link href="//fk-cp-zion/css/Browse.chunk.ff59ed.css" rel="stylesheet
ne at Best Price in India - All Categories | Flipkart.com</title><meta content="Laptop- Buy Proc
All Categories | Flipkart.com" name="og_title" property="og:title"><meta content="Laptop- Buy
```

Product and brand name

In [5]:

```

1 #<div class="_4rR01T">Lenovo Intel Core i3 11th Gen - (8 GB/512 GB SSD,
2 r1 = flip.find('div', {'class': '_4rR01T'})
3 print(r1.text)                                ##product
4 print(r1.text.split()[0])                      ##brand

```

Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Windows 11 Home) 14ITL05 Thin and Light Lap
Lenovo

product review, rating

In [6]:

```

1 #div class="gUuXy-">
2 #<span id="productRating_LSTCOMGBNFUTNYFZRJQFAMIHE_COMGBNFUTNYFZRJQ_" (
3 #<div class="_3LWZLK">4.3<div class="_30jeq3 _1_WHN1">₹39,650</div>
2 #<div class="_3I9_wc _27UcVY">₹59,890</div>
3 #<div class="_3Ay6Sb"><span>33% off</span></div></div>
4 r4 = flip.find('div',{'class': '_30jeq3 _1_WHN1'})
5 print(r4.text)                                #dis_price
6 r5 = flip.find('div',{'class': '_3I9_wc _27UcVY'})
7 print(r5.text)                                #act_price
8 r6 = flip.find('div',{'class': '_3Ay6Sb'})
9 print(r6.text)                                # discount%
```

₹33,890

₹60,890

44% off

Product Details

In [8]:

```
1 p = flip.find('ul',{'class': '_1xgFaf'})
2 print(p.li.text)
3 print(p.li.next_sibling.text.split()[0])
4 print(p.li.next_sibling.next_sibling.text)
5 print(p.li.next_sibling.next_sibling.next_sibling.text)
6 print(p.li.next_sibling.next_sibling.next_sibling.next_sibling.text)
```

Intel Core i3 Processor (11th Gen)

8

64 bit Windows 11 Operating System

256 GB SSD

35.56 cm (14 inch) Display

In [9]:

```

1 product1 = []
2 brand1 = []
3 starrating1 = []
4 num_review1 = []
5 num_rating1 = []
6 dis_rate1 = []
7 act_rate1 = []
8 discount1 = []
9 processor1 = []
10 ram1 = []
11 oper_sys1 = []
12 ssd1 = []
13 display1 = []
14 for i in flip.find_all('a',{'class':'_1fQZEK'}):
15     product1.append(i.find('div',{'class':'_4rR01T'}).text)
16     brand1.append(i.find('div',{'class':'_4rR01T'}).text.split()[0])
17     starrating1.append(i.find('div',{'class':'_3LWZ1K'}).text)
18     num_review1.append(i.find('div',{'class':'gUuXy-'}).text.split()[3])
19     num_rating1.append(i.find('span',{'class':'_2_R_DZ'}).text.split()[1])
20     dis_rate1.append(i.find('div',{'class':'_30jeq3 _1_WHN1'}).text)
21     act_rate1.append(i.find('div',{'class':'_3I9_wc _27UcVY'}).text)
22     discount1.append(i.find('div',{'class':'_3Ay6Sb'}).text)
23     processor1.append(i.find('ul',{'class':'_1xgFaf'}).li.text)
24     ram1.append(i.find('ul',{'class':'_1xgFaf'}).li.next_sibling.text)
25     oper_sys1.append(i.find('ul',{'class':'_1xgFaf'}).li.next_sibling.text)
26     ssd1.append(i.find('ul',{'class':'_1xgFaf'}).li.next_sibling.next_sibling.text)
27     display1.append(i.find('ul',{'class':'_1xgFaf'}).li.next_sibling.next_sibling.text)
28 print(product1)
29 print('-----')
30 print(brand1)
31 print('-----')
32 print(starrating1)
33 print('-----')
34 print(num_review1)
35 print('-----')
36 print(num_rating1)
37 print('-----')
38 print(dis_rate1)

```

```
39 print('-----')
40 print(act_rate1)
41 print('-----')
42 print(discount1)
43 print('-----')
44 print(processor1)
45 print('-----')
46 print(ram1)
47 print('-----')
48 print(oper_sys1)
49 print('-----')
50 print(ssd1)
51 print('-----')
52 print(display1)
```

```
['Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Windows 11 Home) 14ITL05 Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15ITL6 Thin and Light Laptop', 'ASUS TUF Gaming F15 (8 GB/256 GB SSD/Windows 11 Home/4 GB Graphics/NVIDIA GeForce RTX 3050Ti) 15.6 inch Thin and Light Laptop', 'ASUS VivoBook 15 (2022) Core i3 10th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15.6 inch Thin and Light Laptop', 'Lenovo Athlon Dual Core - (4 GB/256 GB SSD/DOS) E41-55 45-2 Laptop', 'acer Aspire 3 Ryzen 3 Dual Core - (8 GB/256 GB SSD/Windows 11 Home) A315-23 Laptop', 'Lenovo IdeaPad Gaming Core i5 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15.6 inch Thin and Light Laptop', 'ASUS VivoBook 14 (2021) Celeron Dual Core - (4 GB/256 GB SSD/Windows 11 Home) X415MA-BV614 Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Windows 10 Home) RMNB1001 Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Windows 11 Home) V15 ITL G2 Laptop', 'Lenovo Legion 5 Pro Ryzen 7 Octa Core 5800H 16 GB Graphics/NVIDIA GeForce RTX 3050Ti 16 inch Thin and Light Laptop', 'ASUS Vivobook 15 Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) X515EA Thin and Light Laptop', 'DELL Inspiron Core i3 11th Gen - (8 GB/1 TB HDD/256 GB SSD/Windows 11 Home) D560841WIN9S Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Mac OS Big Sur) MGN63HN/A', 'Lenovo IdeaPad Gaming 3 Ryzen 7 Octa Core AMD R5-5800H 16 GB Graphics/NVIDIA GeForce RTX 3050Ti 16 inch Thin and Light Laptop', 'MSI Bravo 15 Ryzen 5 Hexa Core AMD R5-5600H - (8 GB/512 GB SSD/Windows 11 Home) B5R Thin and Light Laptop', 'RedmiBook Pro Core i5 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/256 GB SSD/Windows 10 Home) RMNB1001 Thin and Light Laptop', 'Lenovo Intel Core i5 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15.6 inch Thin and Light Laptop', 'HP 15s Intel Core i3 12th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15.6 inch Thin and Light Laptop', 'ASUS VivoBook K15 OLED (2022) Ryzen 5 Hexa Core AMD R5-5500U - (8 GB/1 TB SSD/Windows 11 Home) 15.6 inch Thin and Light Laptop', 'DELL Vostro Core i3 11th Gen - (8 GB/1 TB HDD/256 GB SSD/Windows 11 Home) Vostro 3510 Thin and Light Laptop', 'Lenovo IdeaPad 3 Core i5 11th Gen - (8 GB/512 GB SSD/Windows 11 Home/4 GB Graphics/NVIDIA GeForce RTX 3050Ti) 15.6 inch Thin and Light Laptop']
```

```
['Lenovo', 'Lenovo', 'ASUS', 'HP', 'ASUS', 'Lenovo', 'acer', 'Lenovo', 'ASUS', 'realme', 'Lenovo', 'Lenovo', 'MSI', 'RedmiBook', 'realme', 'Lenovo', 'HP', 'ASUS', 'DELL', 'Lenovo']
```

```
['4.2', '4.2', '4.4', '4.2', '4.3', '3.9', '4.2', '4.3', '4.1', '4.4', '4', '4.7', '4.1', '4.2', '4.3', '4.3', '4.4', '4.1', '4.4']
```

```
['66', '21', '463', '157', '1,040', '3', '72', '227', '95', '1,865', '6', '5', '26', '40', '793', '14', '146', '11', '43']
```

```
['870', '280', '4,733', '1,762', '10,406', '15', '367', '2,634', '790', '12,565', '98', '32', '2,3,918', '12,565', '8', '77', '1,224', '130', '395']
```

```
['₹33,890', '₹51,990', '₹49,990', '₹37,990', '₹32,990', '₹18,990', '₹26,990', '₹51,990', '₹23,990', '₹38,990', '₹42,490', '₹86,990', '₹67,990', '₹47,990', '₹38,990', '₹35,990', '₹68,290', '₹44,000']
```

```
['₹60,890', '₹71,990', '₹70,990', '₹47,206', '₹45,990', '₹38,000', '₹42,999', '₹76,890', '₹33,990', '₹50,990', '₹61,202', '₹99,900', '₹1,02,090', '₹72,990', '₹59,999', '₹54,999', '₹99,290', '₹690']
```

```
['44% off', '27% off', '29% off', '19% off', '28% off', '50% off', '37% off', '32% off', '29% off', '23% off', '30% off', '12% off', '33% off', '34% off', '35% off', '34% off', '31% off', '20% off']
```

```
['Intel Core i3 Processor (11th Gen)', 'Intel Core i5 Processor (11th Gen)', 'Intel Core i5 Processor (11th Gen)', 'Intel Core i3 Processor (10th Gen)', 'AMD Athlon Dual Core Processor', 'AMD Athlon Dual Core Processor (11th Gen)', 'Intel Celeron Dual Core Processor', 'Stylish & Portable Thin and Light Laptop', 'AMD Ryzen 7 Octa Core Processor', 'Intel Core i3 Processor (11th Gen)', 'Intel Core i3 Processor (11th Gen)', 'Intel Core i5 Processor (11th Gen)', 'Intel Core i3 Processor (11th Gen)', 'Intel Core i5 Processor (12th Gen)', 'AMD Ryzen 5 Hexa Core Processor', 'Processor: Intel i3-1115G4 (Base- 1.7 GHz) 8 GB/256 GB SSD/Windows 11 Home', 'Intel Core i5 Processor (11th Gen)']
```

```
['8', '8', '8', '8', '8', '4', '8', '8', '4', '14', '8', '16', '8', 'RAM', '8', '8', '8', '8', '8']
```

```
['64 bit Windows 11 Operating System', '64 bit Windows 11 Operating System', 'Windows 11 Operating System', '64 bit Windows 11 Operating System', 'DOS Operating System', '64 bit Windows 11 Operating System']
```


In [15]:

```
1 print(dis_rate1)
```

```
['₹33,890', '₹51,990', '₹49,990', '₹37,990', '₹32,990', '₹18,990', '₹26,990', '₹51,990', '₹23,990', '₹38,990', '₹42,490', '₹86,990', '₹67,990', '₹47,990', '₹38,990', '₹35,990', '₹68,290', '₹44,000']
```

In [16]:

```
1 dis_rate1 = [i.replace('₹','') for i in dis_rate1]
2 print(dis_rate1)
```

```
['33,890', '51,990', '49,990', '37,990', '32,990', '18,990', '26,990', '51,990', '23,990', '38,990', '42,490', '86,990', '67,990', '47,990', '38,990', '35,990', '68,290', '44,990', '49,990', '39,990']
```

In [17]:

```
1 #Dicount Price
2 dis_rate1 = [i.replace(',','') for i in dis_rate1]
3 print(dis_rate1)
```

```
['33890', '51990', '49990', '37990', '32990', '18990', '26990', '51990', '23990', '35990', '33990', '38990', '42490', '86990', '67990', '47990', '38990', '35990', '68290', '44990', '49990', '39990', '62990']
```

In [18]:

```
1 print(act_rate1)
```

```
['₹60,890', '₹71,990', '₹70,990', '₹47,206', '₹45,990', '₹38,000', '₹42,999', '₹76,890', '₹33,990', '₹54,999', '₹61,202', '₹99,900', '₹1,02,090', '₹72,990', '₹59,999', '₹54,999', '₹99,290', '₹56,260', '₹78,990', '₹58,690']
```

In [19]:

```
1 act_rate1 = [i.replace('₹','') for i in act_rate1]
2 print(act_rate1)
```

```
['60,890', '71,990', '70,990', '47,206', '45,990', '38,000', '42,999', '76,890', '33,990', '54,999', '61,202', '99,900', '1,02,090', '72,990', '59,999', '54,999', '99,290', '56,260', '78,990', '58,690']
```

In [20]:

```
1 #Actual Rate
2 act_rate1 = [i.replace(',','') for i in act_rate1]
3 print(act_rate1)
```

```
['60890', '71990', '70990', '47206', '45990', '38000', '42999', '76890', '33990', '54999', '59760', '102090', '72990', '59999', '54999', '99290', '56260', '78990', '58489', '93690']
```

In [21]:

```
1 print(discount1)
```

```
['44% off', '27% off', '29% off', '19% off', '28% off', '50% off', '37% off', '32% off', '29% of',  
'23% off', '30% off', '12% off', '33% off', '34% off', '35% off', '34% off', '31% off', '20% off']
```

In [22]:

```
1 discount1 = [i.replace('%','') for i in discount1]  
2 print(discount1)
```

```
['44 off', '27 off', '29 off', '19 off', '28 off', '50 off', '37 off', '32 off', '29 off', '34 c',  
'30 off', '12 off', '33 off', '34 off', '35 off', '34 off', '31 off', '20 off', '36 off', '31 of']
```

In [23]:

```
1 discount1 = [i.replace('off','') for i in discount1]  
2 print(discount1)
```

```
['44 ', '27 ', '29 ', '19 ', '28 ', '50 ', '37 ', '32 ', '29 ', '34 ', '43 ', '25 ', '23 ', '30',  
' ', '31 ', '20 ', '36 ', '31 ', '32 ']
```

In [24]:

```
1 print(ram1)
```

```
['8', '8', '8', '8', '8', '4', '8', '8', '4', '14', '8', '16', '8', 'RAM', '8', '8', '8', '8', ' ']
```

In [26]:

1 df = pd.DataFrame({'Product':product1, 'Brands':brand1, 'Star_rating':s

2 'Num_rating':num_rating1, 'Disc_rate':dis_rate1, 'Act_

3 'Processor':processor1, 'RAM(GB)':ram1,'Oper_Sys':oper

4 df.head() # top 5 rows and columns

ands	Star_rating	Num_Review	Num_rating	Disc_rate	Act_rate	Disount%	Processor	I
novo	4.2	66	870	33890	60890	44	Intel Core i3 Processor (11th Gen)	8
novo	4.2	21	280	51990	71990	27	Intel Core i5 Processor (11th Gen)	8
US	4.4	463	4733	49990	70990	29	Intel Core i5 Processor (10th Gen)	8
'	4.2	157	1762	37990	47206	19	Intel Core i3 Processor (11th Gen)	8
US	4.3	1040	10406	32990	45990	28	Intel Core i3 Processor (10th Gen)	8

In [27]:

1 df.shape

(24, 13)

In [28]:

1 df.dtypes

```
Product      object
Brands       object
Star_rating  object
Num_Review   object
Num_rating   object
Disc_rate    object
Act_rate     object
Disount%     object
Processor    object
RAM(GB)      object
Oper_Sys     object
SSD(GB)      object
Display      object
dtype: object
```

In [29]:

1 df.columns

```
Index(['Product', 'Brands', 'Star_rating', 'Num_Review', 'Num_rating',
      'Disc_rate', 'Act_rate', 'Disount%', 'Processor', 'RAM(GB)', 'Oper_Sys',
      'SSD(GB)', 'Display'],
      dtype='object')
```

In [30]:

1 df.isnull().sum()

```
Product      0
Brands       0
Star_rating  0
Num_Review   0
Num_rating   0
Disc_rate    0
Act_rate     0
Disount%     0
Processor    0
RAM(GB)      0
Oper_Sys     0
SSD(GB)      0
Display      0
dtype: int64
```

In [32]:

1 df.duplicated().sum()

```
0
```

```
In [35]: 1 df['Star_rating'] = df['Star_rating'].astype(float)
2 df['Num_Review'] = df['Num_Review'].astype(int)
3 df['Num_rating'] = df['Num_rating'].astype(int)
4 df['Disc_rate'] = df['Disc_rate'].astype(int)
5 df['Act_rate'] = df['Act_rate'].astype(int)
6 df['Disount%'] = df['Disount%'].astype(float)
7
```

```
In [36]: 1 df.dtypes
```

```
Product      object
Brands        object
Star_rating   float64
Num_Review    int32
Num_rating    int32
Disc_rate     int32
Act_rate      int32
Disount%      float64
Processor     object
RAM(GB)       object
Oper_Sys      object
SSD(GB)       object
Display       object
dtype: object
```

```
In [37]: 1 df.to_csv('Flipkart.csv')
```

```
In [38]: 1 df.columns
```

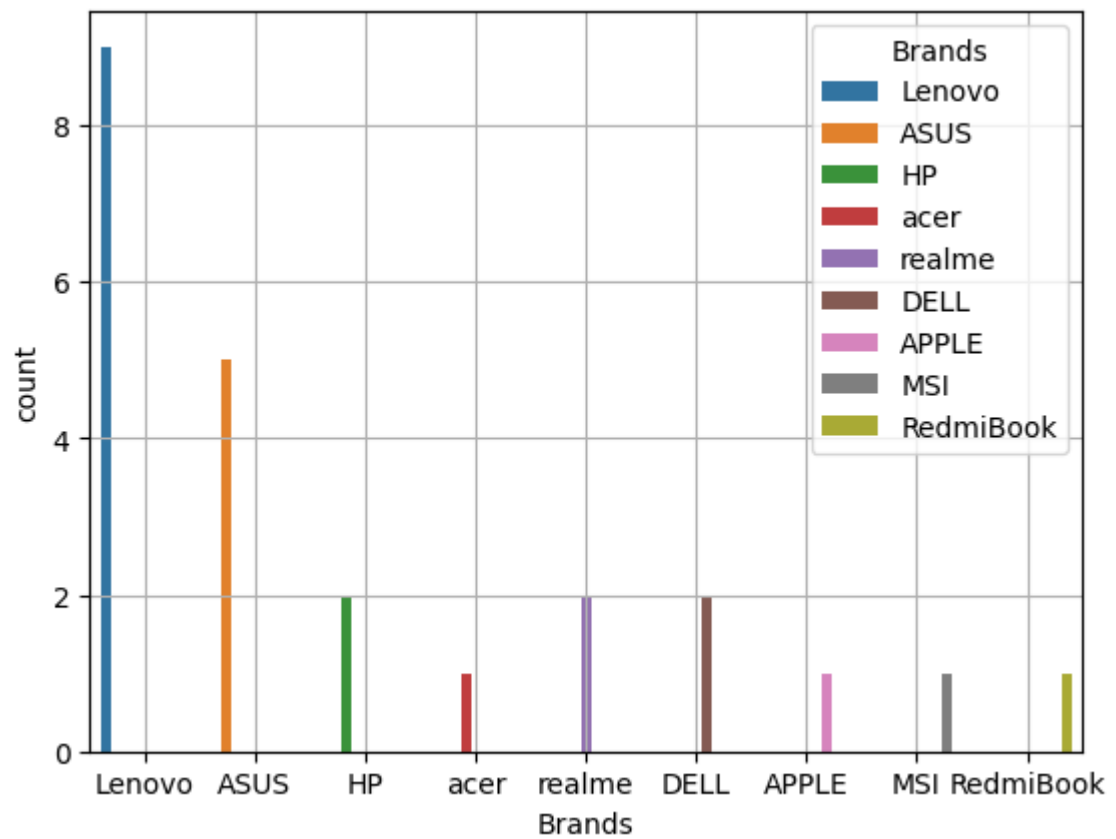
```
Index(['Product', 'Brands', 'Star_rating', 'Num_Review', 'Num_rating',
       'Disc_rate', 'Act_rate', 'Disount%', 'Processor', 'RAM(GB)', 'Oper_Sys',
       'SSD(GB)', 'Display'],
      dtype='object')
```

```
In [39]: 1 df['Brands'].value_counts()
```

```
Lenovo      9
ASUS        5
HP          2
realme      2
DELL        2
acer        1
APPLE       1
MSI         1
RedmiBook   1
Name: Brands, dtype: int64
```

In [40]:

```
1 sns.countplot(x = 'Brands', data = df, hue = 'Brands')  
2 plt.grid()
```



```
In [41]: 1 p1 = df.groupby('Brands')['Act_rate', 'Disc_rate'].agg(['max'])
        2 p1
```

C:\Users\AASIF HP\AppData\Local\Temp\ipykernel_10892\41030044.py:1: FutureWarning: Indexing with to a tuple of keys) will be deprecated, use a list instead.

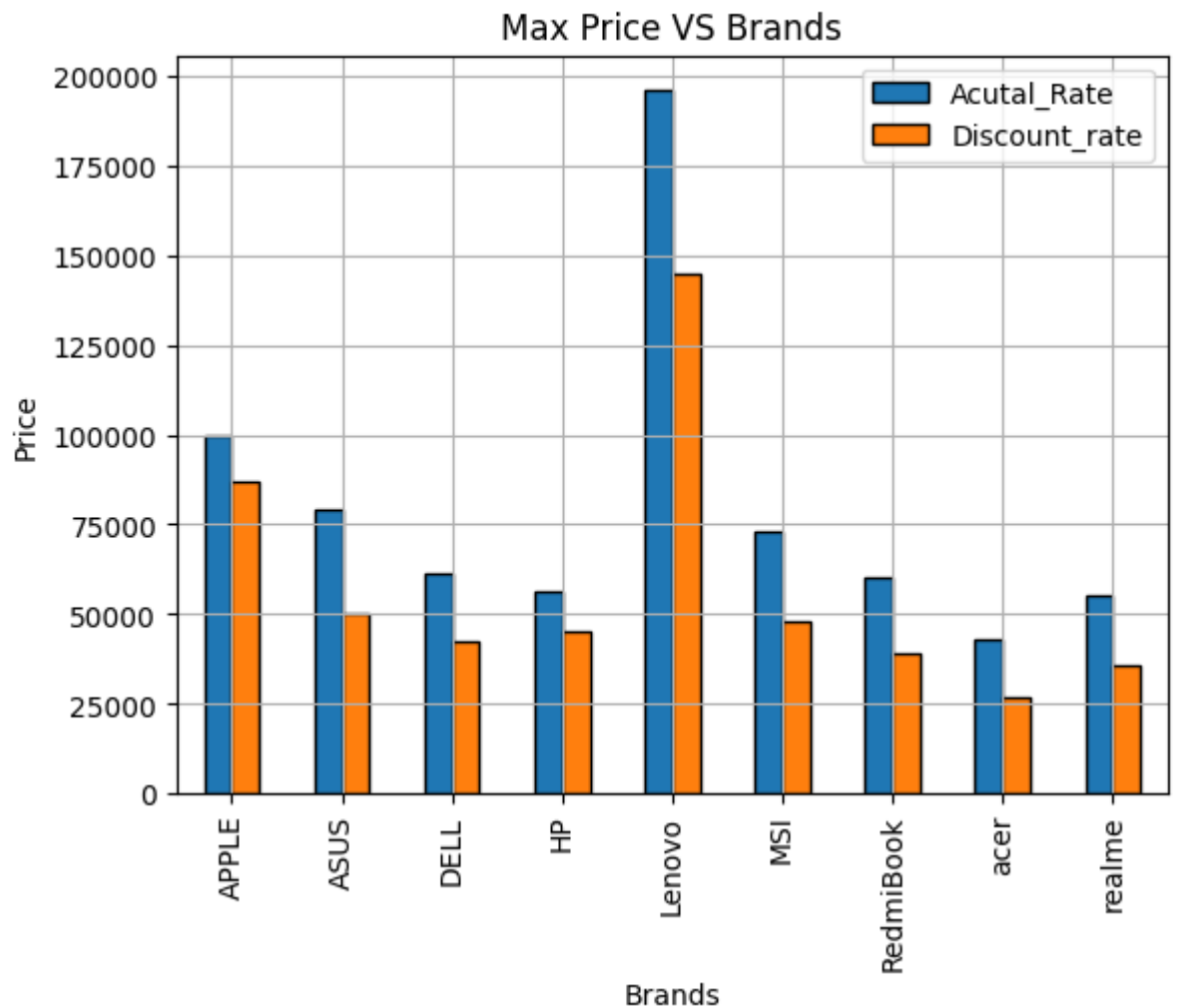
```
p1 = df.groupby('Brands')['Act_rate', 'Disc_rate'].agg(['max'])
```

	Act_rate	Disc_rate
	max	max
Brands		
APPLE	99900	86990
ASUS	78990	49990
DELL	61202	42490
HP	56260	44990
Lenovo	195890	144990
MSI	72990	47990
RedmiBook	59999	38990
acer	42999	26990
realme	54999	35990

In [42]:

```
1 plt.figure(figsize = (9,4))
2 p1.plot(kind = 'bar', edgecolor = 'black')
3 plt.title('Max Price VS Brands')
4 plt.ylabel('Price')
5 plt.legend(['Acutal_Rate', 'Discount_rate'])
6 plt.grid()
```

<Figure size 900x400 with 0 Axes>



In [43]:

```
1 p2 = df.groupby('Brands')['Act_rate', 'Disc_rate'].agg(['min'])
2 p2
```

C:\Users\AASIF HP\AppData\Local\Temp\ipykernel_10892\3858546362.py:1: FutureWarning: Indexing with a tuple of keys will be deprecated, use a list instead.

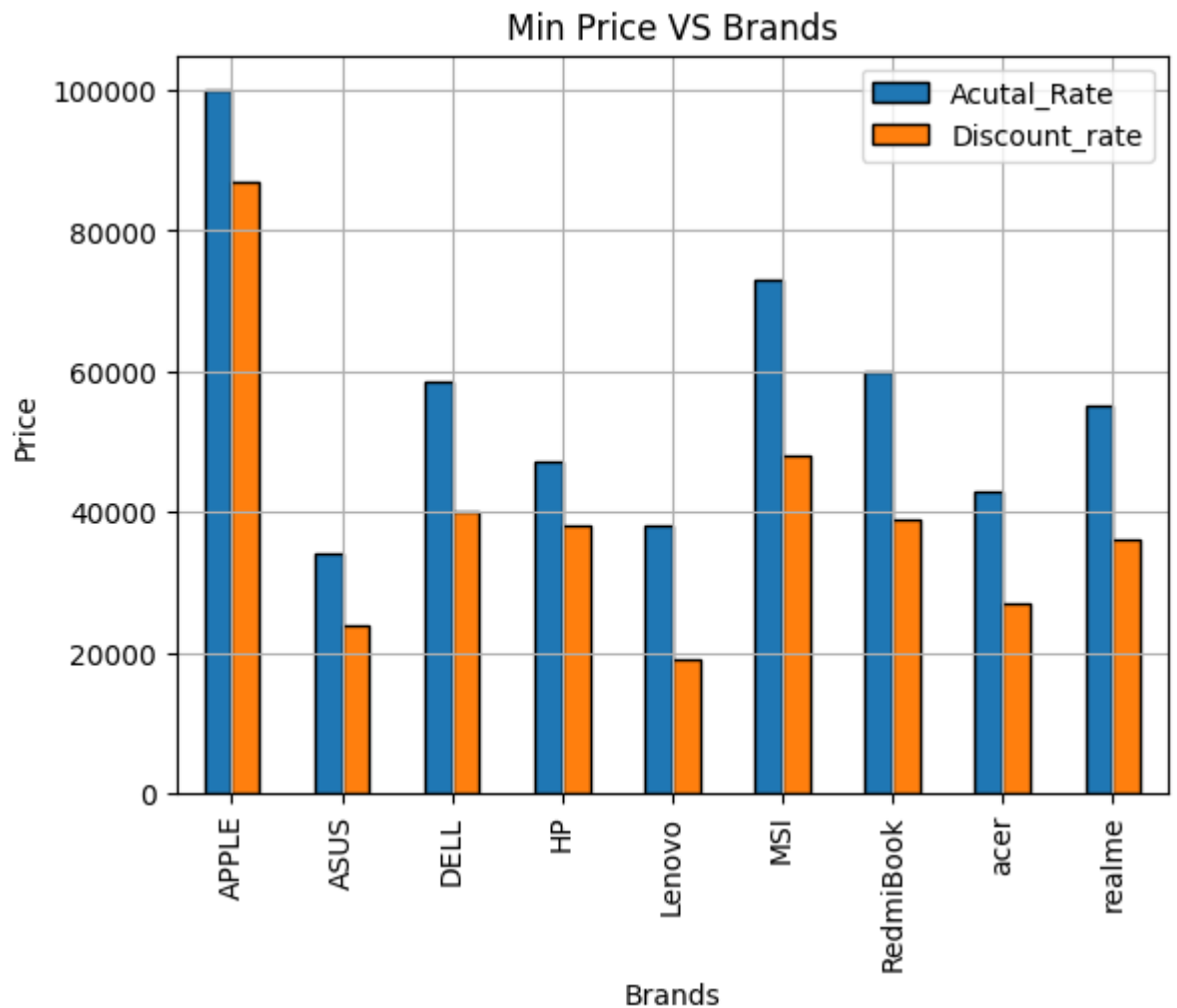
```
p2 = df.groupby('Brands')['Act_rate', 'Disc_rate'].agg(['min'])
```

Brands	Act_rate	Disc_rate
	min	min
APPLE	99900	86990
ASUS	33990	23990
DELL	58489	39990
HP	47206	37990
Lenovo	38000	18990
MSI	72990	47990
RedmiBook	59999	38990
acer	42999	26990
realme	54999	35990

In [44]:

```
1 plt.figure(figsize = (9,4))
2 p2.plot(kind = 'bar', edgecolor = 'black')
3 plt.title('Min Price VS Brands')
4 plt.ylabel('Price')
5 plt.legend(['Acutal_Rate', 'Discount_rate'])
6 plt.grid()
```

<Figure size 900x400 with 0 Axes>



In [45]:

```
1 p3 = df.groupby(['Brands'])['Num_Review','Num_rating'].agg(['min'])
2 p3
```

C:\Users\AASIF HP\AppData\Local\Temp\ipykernel_10892\2914613025.py:1: FutureWarning: Indexing with a tuple of keys) will be deprecated, use a list instead.

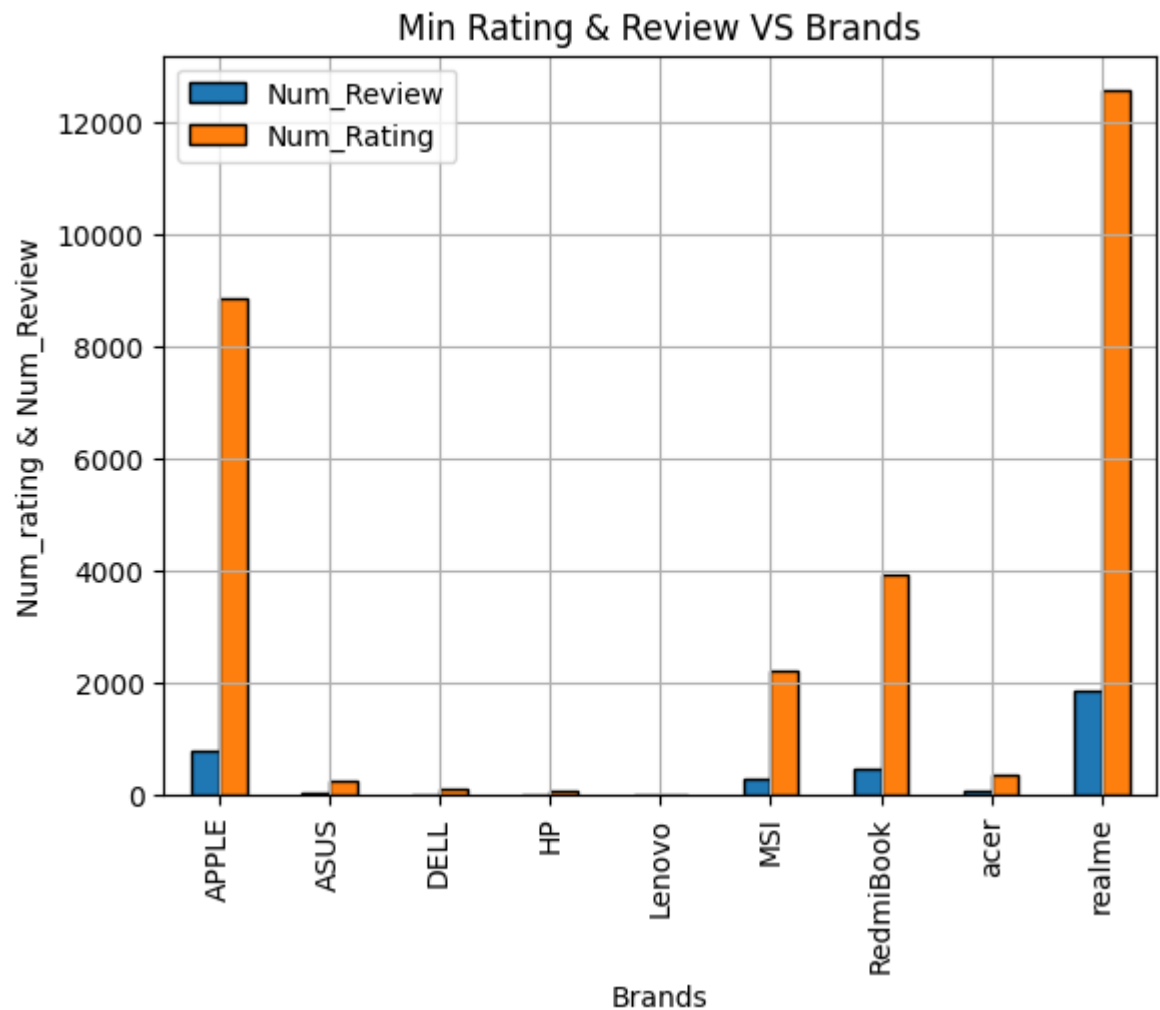
```
p3 = df.groupby(['Brands'])['Num_Review','Num_rating'].agg(['min'])
```

Brands	Num_Review	Num_rating
	min	min
APPLE	793	8843
ASUS	26	257
DELL	11	130
HP	14	77
Lenovo	0	8
MSI	297	2224
RedmiBook	459	3918
acer	72	367
realme	1865	12565

In [46]:

```
1 plt.figure(figsize = (9,4))
2 p3.plot(kind = 'bar', edgecolor = 'black')
3 plt.title('Min Rating & Review VS Brands')
4 plt.ylabel('Num_rating & Num_Review')
5 plt.legend(['Num_Review', 'Num_Rating'])
6 plt.grid()
```

<Figure size 900x400 with 0 Axes>



In [47]:

```
1 p4 = df.groupby(['Brands'])['Num_Review','Num_rating'].agg(['max'])
2 p4
```

C:\Users\AASIF HP\AppData\Local\Temp\ipykernel_10892\3634886127.py:1: FutureWarning: Indexing with a tuple of keys) will be deprecated, use a list instead.

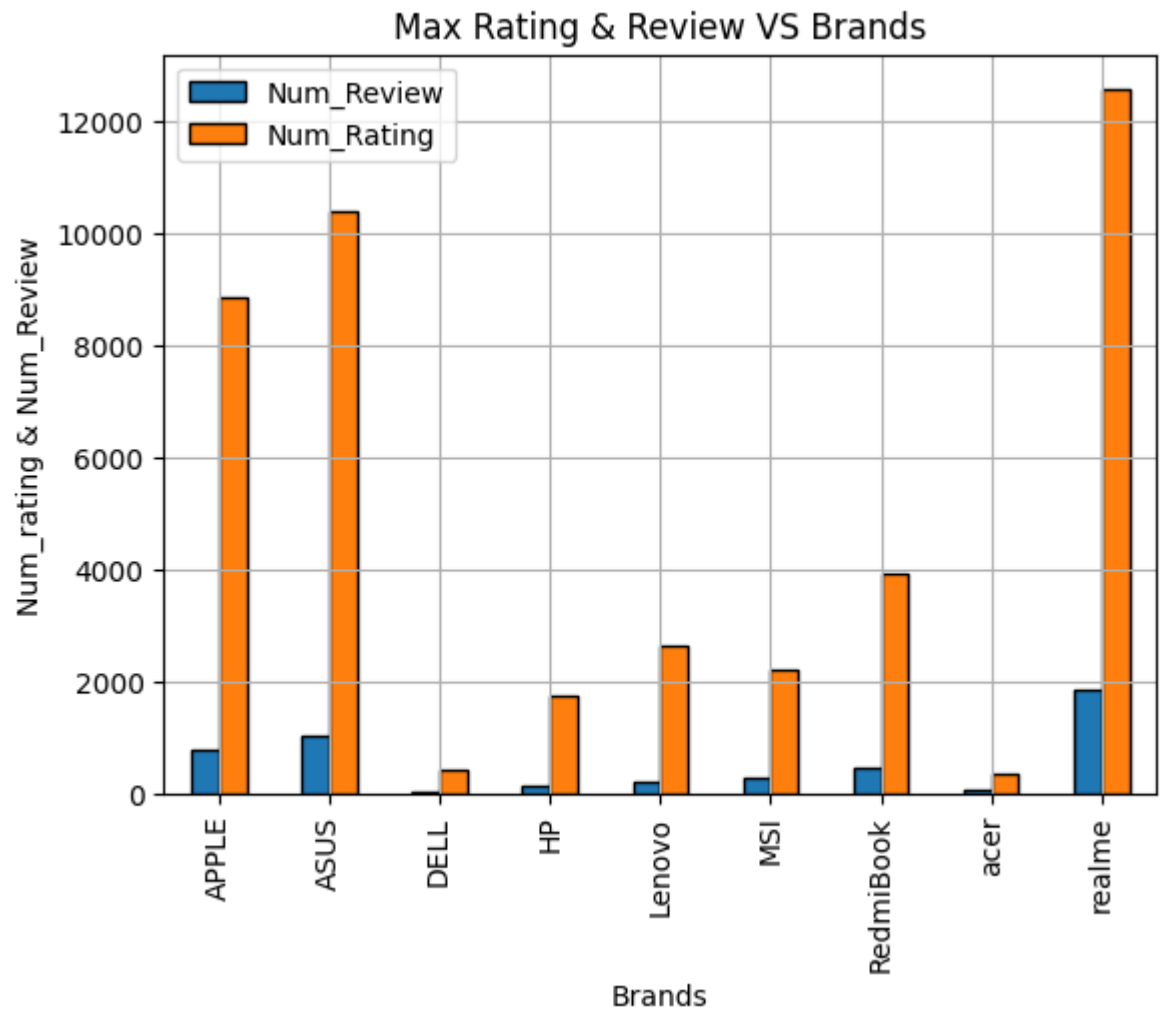
```
p4 = df.groupby(['Brands'])['Num_Review','Num_rating'].agg(['max'])
```

	Num_Review	Num_rating
	max	max
Brands		
APPLE	793	8843
ASUS	1040	10406
DELL	40	440
HP	157	1762
Lenovo	227	2634
MSI	297	2224
RedmiBook	459	3918
acer	72	367
realme	1865	12565

In [48]:

```
1 plt.figure(figsize = (9,4))
2 p4.plot(kind = 'bar', edgecolor = 'black')
3 plt.title('Max Rating & Review VS Brands')
4 plt.ylabel('Num_rating & Num_Review')
5 plt.legend(['Num_Review', 'Num_Rating'])
6 plt.grid()
7 plt.show()
```

<Figure size 900x400 with 0 Axes>



In [49]:

```
1 p5 = df.groupby(['Brands'])['Star_rating', 'Disount%'].agg(['max'])
2 p5
```

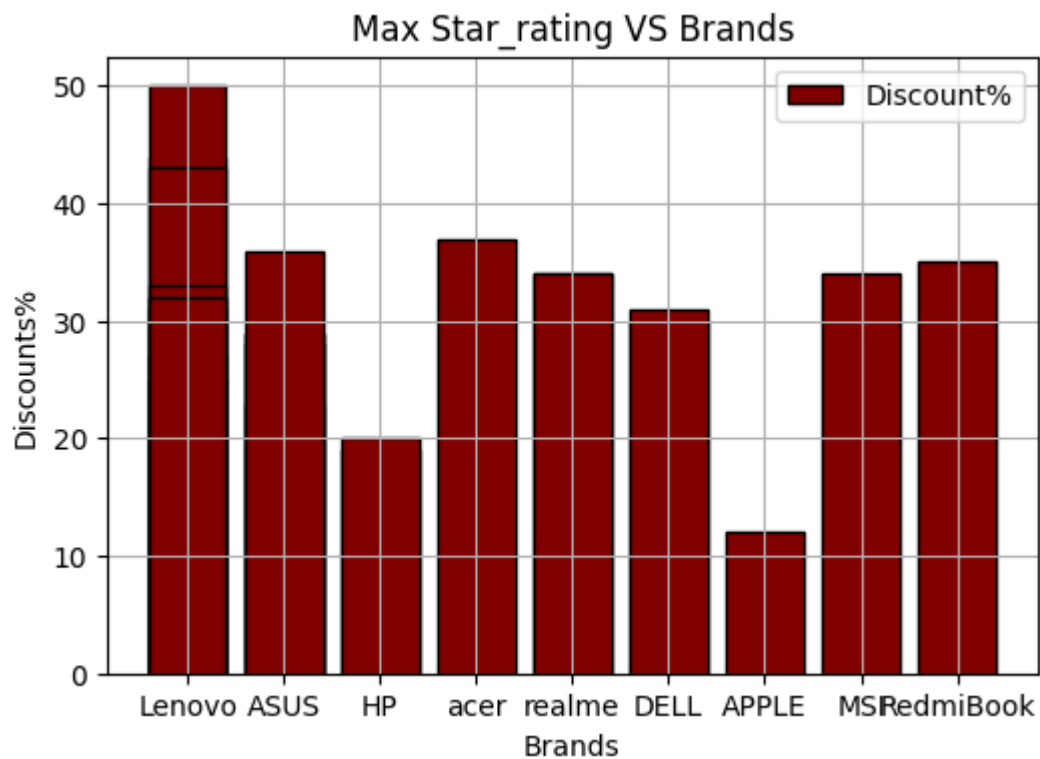
C:\Users\AASIF HP\AppData\Local\Temp\ipykernel_10892\647501289.py:1: FutureWarning: Indexing with a tuple of keys) will be deprecated, use a list instead.

```
p5 = df.groupby(['Brands'])['Star_rating', 'Disount%'].agg(['max'])
```

	Star_rating	Disount%
	max	max
Brands		
APPLE	4.7	12.0
ASUS	4.4	36.0
DELL	4.2	31.0
HP	4.3	20.0
Lenovo	4.7	50.0
MSI	4.4	34.0
RedmiBook	4.1	35.0
acer	4.2	37.0
realme	4.4	34.0

In [50]:

```
1 plt.figure(figsize = (6,4))
2 plt.bar(df['Brands'], df['Discount%'], edgecolor = 'black', color = 'mar
3 plt.title('Max Star_rating VS Brands')
4 plt.legend(['Discount%'])
5 plt.xlabel('Brands')
6 plt.ylabel('Discounts%')
7 plt.grid()
8 plt.show()
```



In []:

1

In []:

1

In []:

1

In []:

1

In []:

1

In []:	1	
In []:	1	
In []:	1	