

```
In [1]: import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [3]: import IPython
print(IPython.sys_info())

{'commit_hash': '223e783c4',
 'commit_source': 'installation',
 'default_encoding': 'cp1252',
 'ipython_path': 'C:\\\\Users\\\\PREDATOR\\\\anaconda3\\\\Lib\\\\site-packages\\\\IPython',
 'ipython_version': '7.19.0',
 'os_name': 'nt',
 'platform': 'Windows-10-10.0.19041-SP0',
 'sys_executable': 'C:\\\\Users\\\\PREDATOR\\\\anaconda3\\\\python.exe',
 'sys_platform': 'win32',
 'sys_version': '3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit '
 '(AMD64)]'}
```

```
In [13]: app_data = pd.read_csv('C:/Users/PREDATOR/Downloads/AppleStore.csv', index_col=0)
```

```
In [10]: type(app_data)
```

```
Out[10]: pandas.core.frame.DataFrame
```

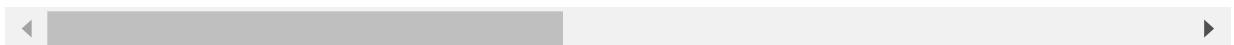
```
In [7]: app_data.head()
```

	Unnamed: 0	id	track_name	size_bytes	currency	price	rating_count_tot	rating_count_ve
0	1	281656475	PAC-MAN Premium	100788224	USD	3.99	21292	26%
1	2	281796108	Evernote - stay organized	158578688	USD	0.00	161065	26%
2	3	281940292	WeatherBug - Local Weather, Radar, Maps, Alerts	100524032	USD	0.00	188583	282%
3	4	282614216	eBay: Best App to Buy, Sell, Save! Online Shop...	128512000	USD	0.00	262241	64%
4	5	282935706	Bible	92774400	USD	0.00	985920	532%

```
In [8]: app_data.tail()
```

	Unnamed: 0	id	track_name	size_bytes	currency	price	rating_count_tot	rating_cour
7192	11081	1187617475	Kubik	126644224	USD	0.00	142	
7193	11082	1187682390	VR Roller- Coaster	120760320	USD	0.00	30	

	Unnamed: 0								
7194	11087	1187779532	Bret Michaels Emojis + Lyric Keyboard	111322112	USD	1.99			15
7195	11089	1187838770	VR Roller Coaster World - Virtual Reality	97235968	USD	0.00			85
7196	11097	1188375727	Escape the Sweet Shop Series	90898432	USD	0.00			3



In [11]: `app_data.shape`

Out[11]: (7197, 17)

In [14]: `app_data.head()` #after removing the index column

1	281656475	PAC-MAN Premium	100788224	USD	3.99		21292		26
2	281796108	Evernote - stay organized	158578688	USD	0.00		161065		26
3	281940292	WeatherBug - Local Weather, Radar, Maps, Alerts	100524032	USD	0.00		188583		2822
4	282614216	eBay: Best App to Buy, Sell, Save! Online Shop...	128512000	USD	0.00		262241		649
5	282935706	Bible	92774400	USD	0.00		985920		5320



In [16]: `app_data.describe()`

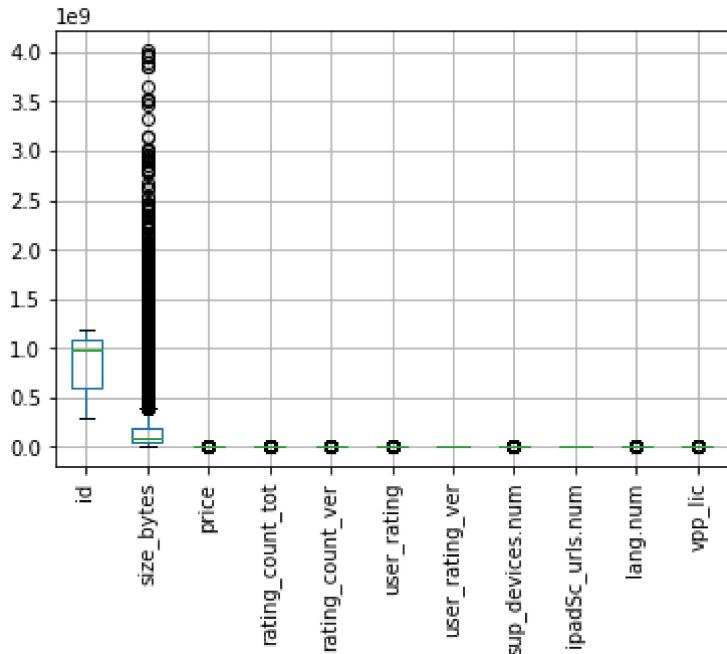
	id	size_bytes	price	rating_count_tot	rating_count_ver	user_rating	us
count	7.197000e+03	7.197000e+03	7197.000000	7.197000e+03	7197.000000	7197.000000	
mean	8.631310e+08	1.991345e+08	1.726218	1.289291e+04	460.373906	3.526956	
std	2.712368e+08	3.592069e+08	5.833006	7.573941e+04	3920.455183	1.517948	
min	2.816565e+08	5.898240e+05	0.000000	0.000000e+00	0.000000	0.000000	
25%	6.000937e+08	4.692275e+07	0.000000	2.800000e+01	1.000000	3.500000	

	id	size_bytes	price	rating_count_tot	rating_count_ver	user_rating	us
50%	9.781482e+08	9.715302e+07	0.000000	3.000000e+02	23.000000	4.000000	
75%	1.082310e+09	1.819249e+08	1.990000	2.793000e+03	140.000000	4.500000	
max	1.188376e+09	4.025970e+09	299.990000	2.974676e+06	177050.000000	5.000000	



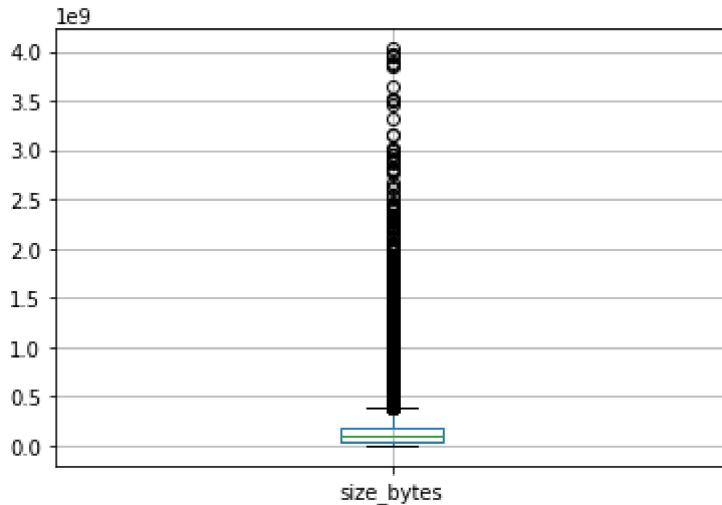
In [19]: `app_data.boxplot()
plt.xticks(rotation=90)`

Out[19]: (array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]),
[Text(1, 0, 'id'),
Text(2, 0, 'size_bytes'),
Text(3, 0, 'price'),
Text(4, 0, 'rating_count_tot'),
Text(5, 0, 'rating_count_ver'),
Text(6, 0, 'user_rating'),
Text(7, 0, 'user_rating_ver'),
Text(8, 0, 'sup_devices.num'),
Text(9, 0, 'ipadSc_urls.num'),
Text(10, 0, 'lang.num'),
Text(11, 0, 'vpp_lic')])



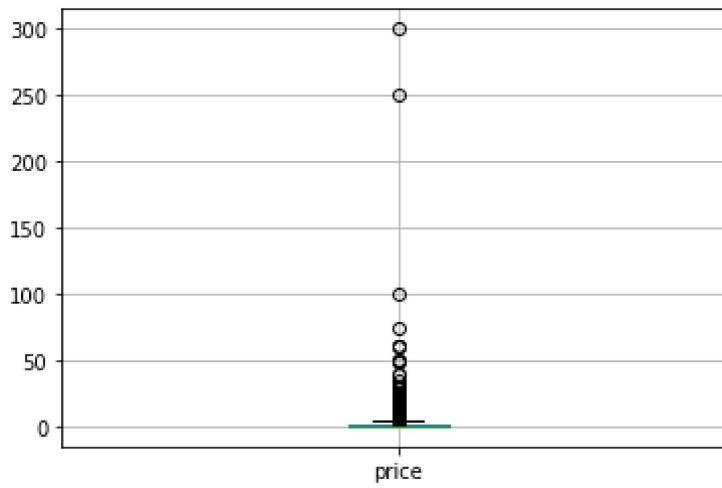
In [45]: `app_data.boxplot(column='size_bytes')`

Out[45]: <AxesSubplot:>



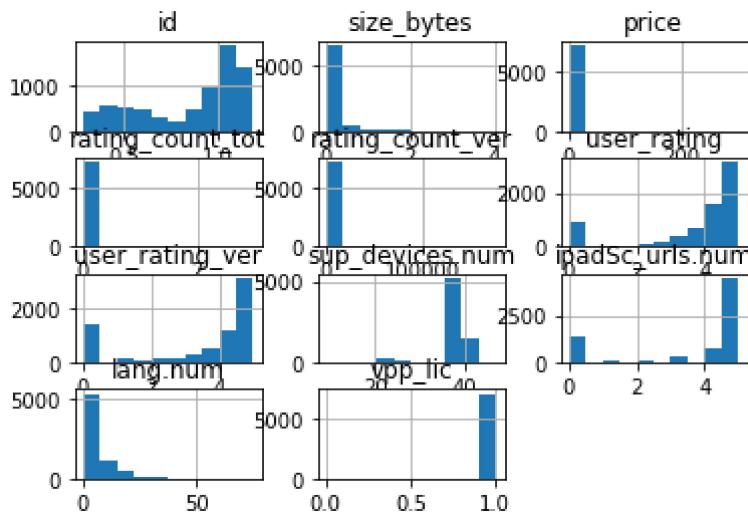
```
In [46]: app_data.boxplot(column='price')
```

```
Out[46]: <AxesSubplot:>
```



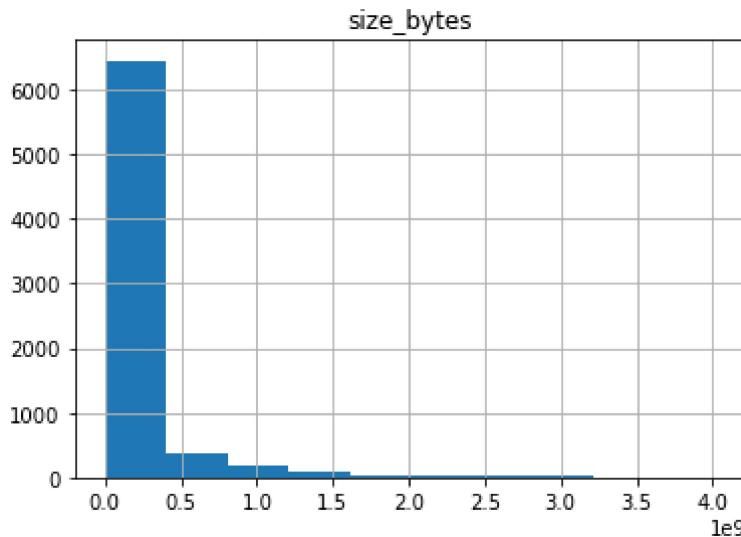
```
In [30]: app_data.hist()
```

```
Out[30]: array([[<AxesSubplot:title={'center':'id'}>,
   <AxesSubplot:title={'center':'size_bytes'}>,
   <AxesSubplot:title={'center':'price'}>],
  [<AxesSubplot:title={'center':'rating_count_tot'}>,
   <AxesSubplot:title={'center':'rating_count_ver'}>,
   <AxesSubplot:title={'center':'user_rating'}>],
  [<AxesSubplot:title={'center':'user_rating_ver'}>,
   <AxesSubplot:title={'center':'sup_devices.num'}>,
   <AxesSubplot:title={'center':'ipadSc_urls.num'}>],
  [<AxesSubplot:title={'center':'lang.num'}>,
   <AxesSubplot:title={'center':'vpp_lic'}>], <AxesSubplot:>]],
 dtype=object)
```



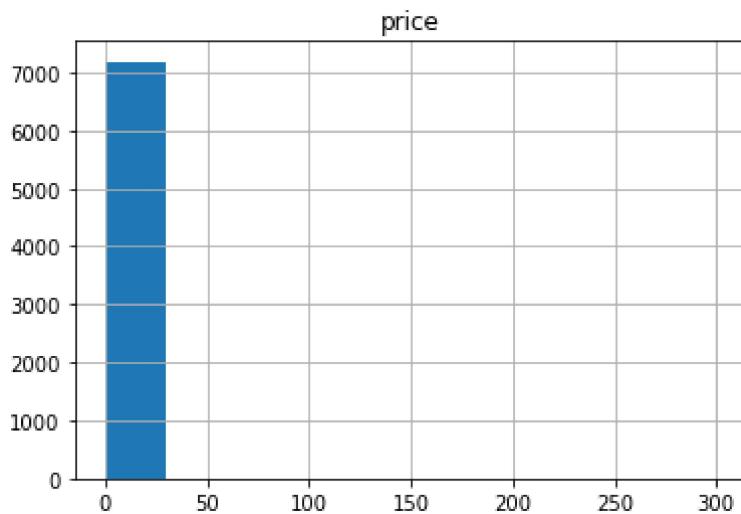
```
In [31]: app_data.hist(column='size_bytes')
```

```
Out[31]: array([[[<AxesSubplot:title={'center':'size_bytes'}>]], dtype=object)
```



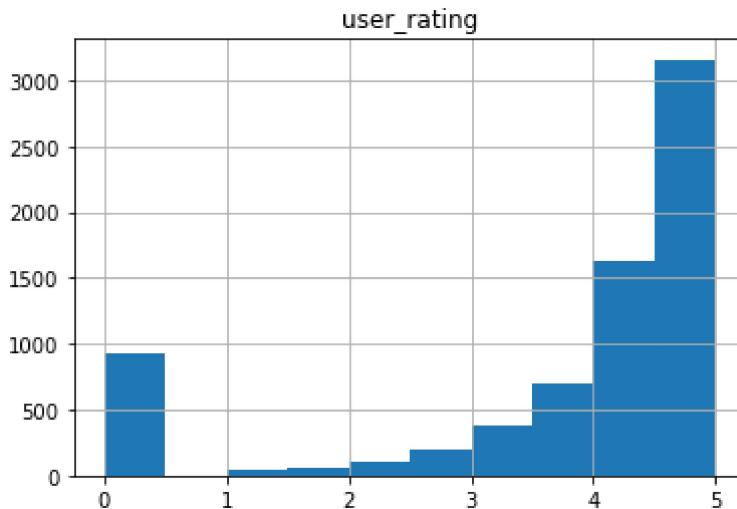
```
In [37]: app_data.hist(column='price')
```

```
Out[37]: array([[[<AxesSubplot:title={'center':'price'}>]], dtype=object)
```



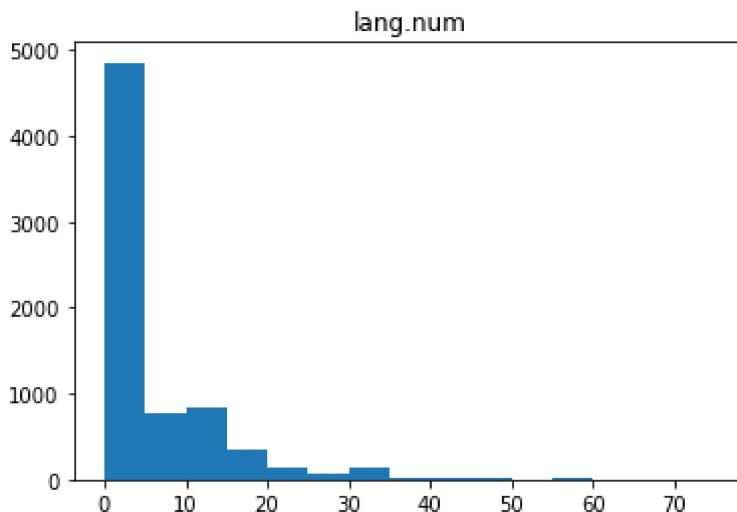
```
In [34]: app_data.hist(column='user_rating')
```

```
Out[34]: array([[[<AxesSubplot:title={'center':'user_rating'}>]], dtype=object)
```



```
In [41]: app_data.hist(column='lang.num' , grid=False, bins=15)
```

```
Out[41]: array([[[<AxesSubplot:title={'center':'lang.num'}>]], dtype=object)
```



```
In [42]: app_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7197 entries, 1 to 11097
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   id               7197 non-null    int64  
 1   track_name       7197 non-null    object  
 2   size_bytes       7197 non-null    int64  
 3   currency         7197 non-null    object  
 4   price            7197 non-null    float64 
 5   rating_count_tot 7197 non-null    int64  
 6   rating_count_ver 7197 non-null    int64  
 7   user_rating      7197 non-null    float64 
 8   user_rating_ver  7197 non-null    float64 
 9   ver              7197 non-null    object  
 10  cont_rating      7197 non-null    object  
 11  prime_genre      7197 non-null    object  
 12  sup_devices.num  7197 non-null    int64  
 13  ipadSc_urls.num 7197 non-null    int64  
 14  lang.num          7197 non-null    int64  
 15  vpp_lic           7197 non-null    int64  
dtypes: float64(3), int64(8), object(5)
memory usage: 955.9+ KB
```

```
In [43]: app_data.size
```

```
Out[43]: 115152
```

```
In [44]: app_data.shape
```

```
Out[44]: (7197, 16)
```

```
In [47]: app_data.isnull().sum()
```

```
Out[47]: id          0
track_name      0
size_bytes      0
currency        0
price           0
rating_count_tot 0
rating_count_ver 0
user_rating      0
user_rating_ver 0
ver              0
cont_rating      0
prime_genre      0
sup_devices.num 0
ipadSc_urls.num 0
lang.num         0
vpp_lic          0
dtype: int64
```

```
In [ ]: # no missing values
```

```
In [48]: app_data[app_data.user_rating>5]
```

```
Out[48]: id track_name size_bytes currency price rating_count_tot rating_count_ver user_rating user_ra
```



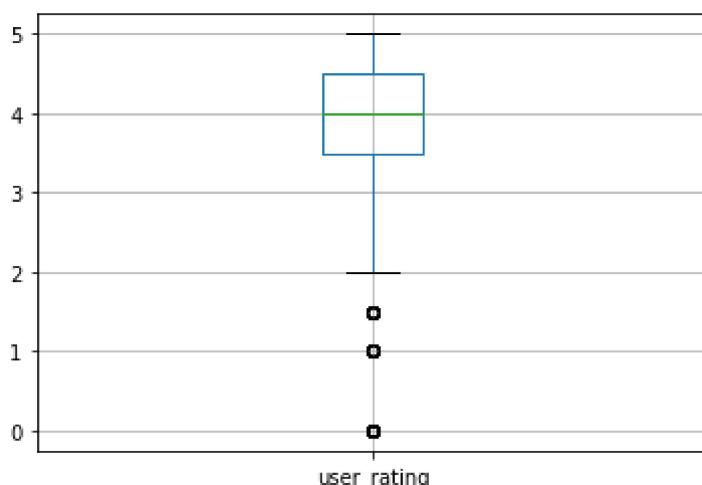
```
In [49]: app_data[app_data.user_rating_ver>5]
```

```
Out[49]: id track_name size_bytes currency price rating_count_tot rating_count_ver user_rating user_ra
```



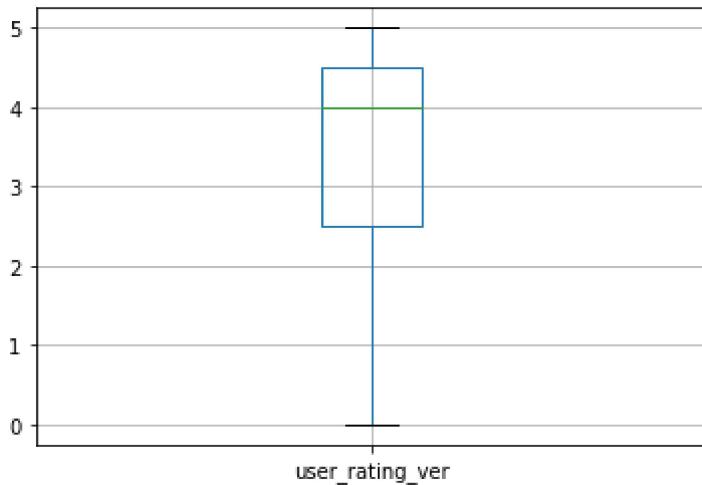
```
In [50]: app_data.boxplot(column='user_rating')
```

```
Out[50]: <AxesSubplot:>
```



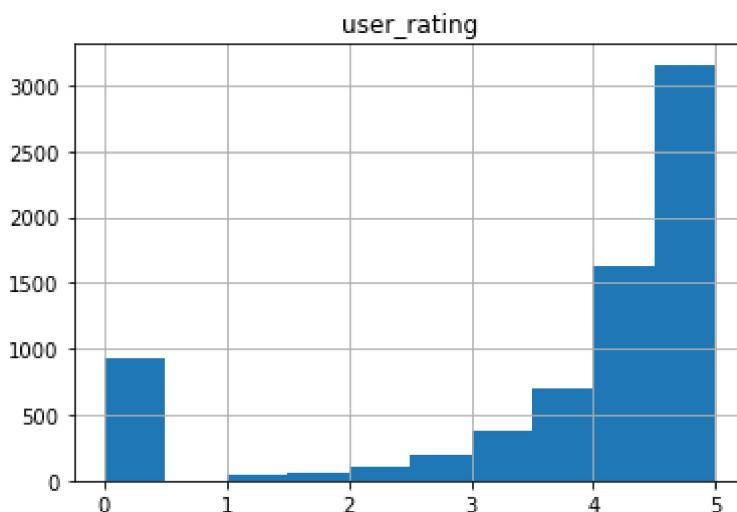
```
In [51]: app_data.boxplot(column='user_rating_ver')
```

```
Out[51]: <AxesSubplot:>
```



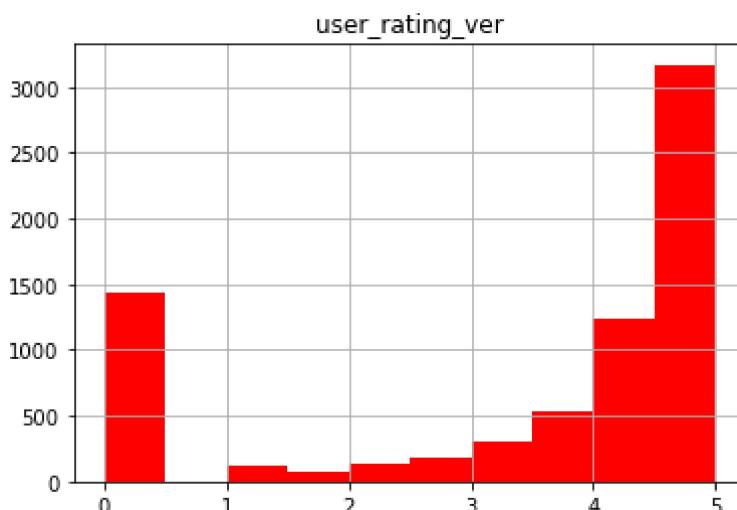
```
In [52]: app_data.hist(column='user_rating')
```

```
Out[52]: array([[[<AxesSubplot:title={'center':'user_rating'}>]]], dtype=object)
```



```
In [55]: app_data.hist(column='user_rating_ver', color='red')
```

```
Out[55]: array([[[<AxesSubplot:title={'center':'user_rating_ver'}>]]], dtype=object)
```



```
In [56]: app_data['cont_rating'] = app_data['cont_rating'].apply(lambda x: str(x).replace('+', '0'))
```

```
In [57]: app_data.head()
```

Out[57]:

	id	track_name	size_bytes	currency	price	rating_count_tot	rating_count_ver	user_rating
1	281656475	PAC-MAN Premium	100788224	USD	3.99	21292	26	4
2	281796108	Evernote - stay organized	158578688	USD	0.00	161065	26	4
3	281940292	WeatherBug - Local Weather, Radar, Maps, Alerts	100524032	USD	0.00	188583	2822	3
4	282614216	eBay: Best App to Buy, Sell, Save! Online Shop...	128512000	USD	0.00	262241	649	4
5	282935706	Bible	92774400	USD	0.00	985920	5320	4



In [60]: app_data.describe()

Out[60]:

	id	size_bytes	price	rating_count_tot	rating_count_ver	user_rating	us
count	7.197000e+03	7.197000e+03	7197.000000	7.197000e+03	7197.000000	7197.000000	
mean	8.631310e+08	1.991345e+08	1.726218	1.289291e+04	460.373906	3.526956	
std	2.712368e+08	3.592069e+08	5.833006	7.573941e+04	3920.455183	1.517948	
min	2.816565e+08	5.898240e+05	0.000000	0.000000e+00	0.000000	0.000000	
25%	6.000937e+08	4.692275e+07	0.000000	2.800000e+01	1.000000	3.500000	
50%	9.781482e+08	9.715302e+07	0.000000	3.000000e+02	23.000000	4.000000	
75%	1.082310e+09	1.819249e+08	1.990000	2.793000e+03	140.000000	4.500000	
max	1.188376e+09	4.025970e+09	299.990000	2.974676e+06	177050.000000	5.000000	



```
In [61]: grp=app_data.groupby('prime_genre')
x=grp['user_rating'].agg(np.mean)
y=grp['user_rating_ver'].agg(np.mean)
z=grp['price'].agg(np.sum)
a=grp['sup_devices.num'].agg(np.sum)
b=grp['ipadSc_urls.num'].agg(np.mean)
print(x)
print(y)
print(z)
print(a)
print(b)
```

prime_genre	
Book	2.477679
Business	3.745614
Catalogs	2.100000
Education	3.376380
Entertainment	3.246729
Finance	2.432692
Food & Drink	3.182540

Games	3.685008
Health & Fitness	3.700000
Lifestyle	2.805556
Medical	3.369565
Music	3.978261
Navigation	2.684783
News	2.980000
Photo & Video	3.800860
Productivity	4.005618
Reference	3.453125
Shopping	3.540984
Social Networking	2.985030
Sports	2.982456
Travel	3.376543
Utilities	3.278226
Weather	3.597222
Name: user_rating, dtype: float64	
prime_genre	
Book	2.111607
Business	3.333333
Catalogs	2.100000
Education	2.952539
Entertainment	2.900000
Finance	1.634615
Food & Drink	2.412698
Games	3.545184
Health & Fitness	3.369444
Lifestyle	2.218750
Medical	3.260870
Music	3.811594
Navigation	2.152174
News	2.213333
Photo & Video	3.485673
Productivity	3.733146
Reference	3.093750
Shopping	2.618852
Social Networking	2.482036
Sports	2.548246
Travel	2.820988
Utilities	2.862903
Weather	3.222222
Name: user_rating_ver, dtype: float64	
prime_genre	
Book	200.54
Business	291.63
Catalogs	7.99
Education	1824.79
Entertainment	475.99
Finance	43.80
Food & Drink	97.80
Games	5533.95
Health & Fitness	344.96
Lifestyle	127.50
Medical	201.85
Music	667.29
Navigation	189.74
News	38.83
Photo & Video	514.18
Productivity	770.84
Reference	309.56
Shopping	1.99
Social Networking	56.76
Sports	108.65
Travel	90.75
Utilities	408.61
Weather	115.59
Name: price, dtype: float64	
prime_genre	
Book	4205

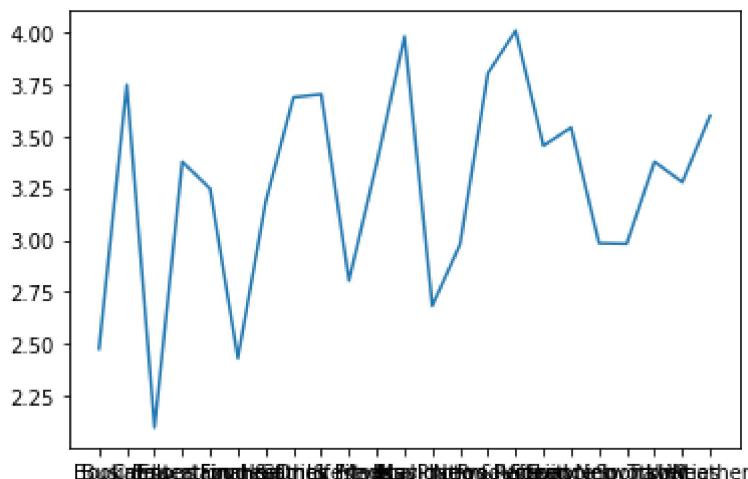
```

Business           2059
Catalogs          373
Education         16614
Entertainment    19619
Finance           3831
Food & Drink     2326
Games             146827
Health & Fitness 6460
Lifestyle          5338
Medical            843
Music              4895
Navigation         1667
News               2742
Photo & Video     12828
Productivity       6415
Reference          2340
Shopping           4468
Social Networking  6099
Sports              4209
Travel              2998
Utilities           9098
Weather             2639
Name: sup_devices.num, dtype: int64
prime_genre
Book               2.991071
Business           3.368421
Catalogs           3.400000
Education          4.421634
Entertainment      3.050467
Finance            1.721154
Food & Drink       2.063492
Games              4.298032
Health & Fitness   2.450000
Lifestyle          1.847222
Medical             3.173913
Music              3.637681
Navigation          2.565217
News               2.760000
Photo & Video      2.957020
Productivity        3.808989
Reference          3.906250
Shopping            2.172131
Social Networking   1.742515
Sports              3.000000
Travel              2.592593
Utilities           2.725806
Weather             3.125000
Name: ipadSc_urls.num, dtype: float64

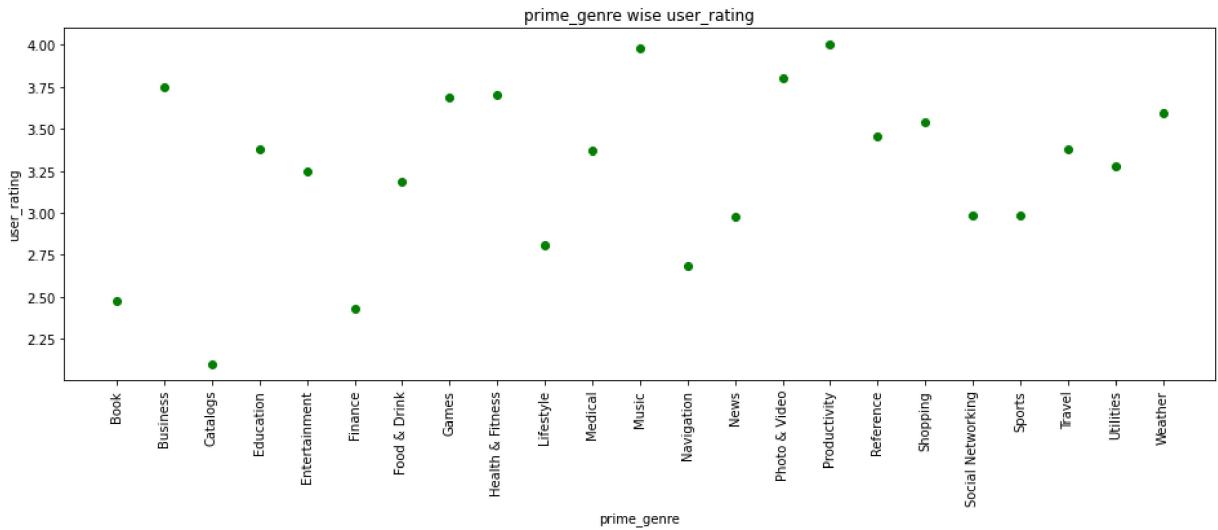
```

In [62]: `plt.plot(x)`

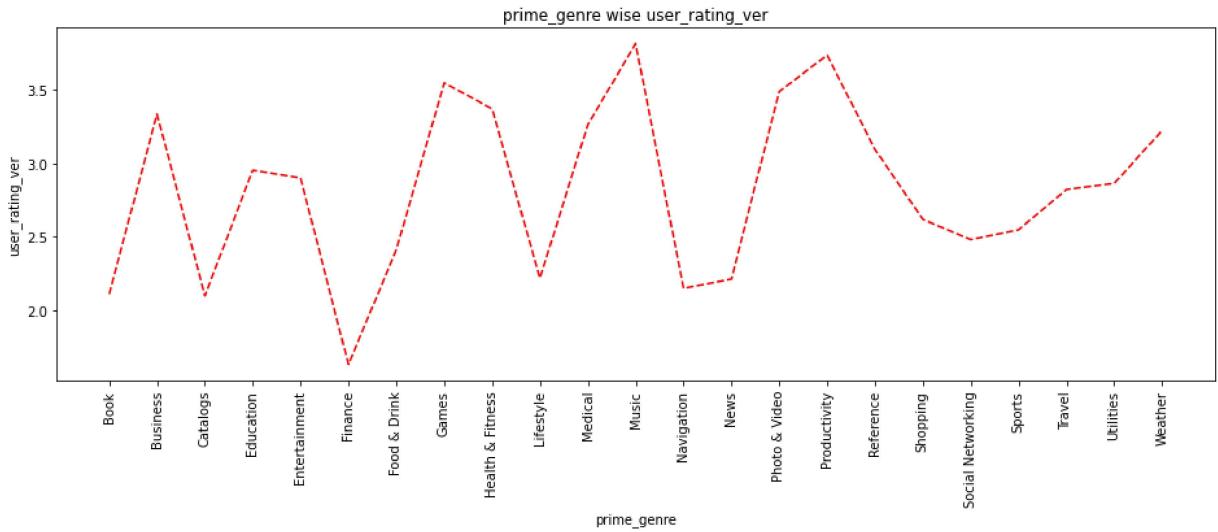
Out[62]: [`<matplotlib.lines.Line2D at 0x21e6f34cb20>`]



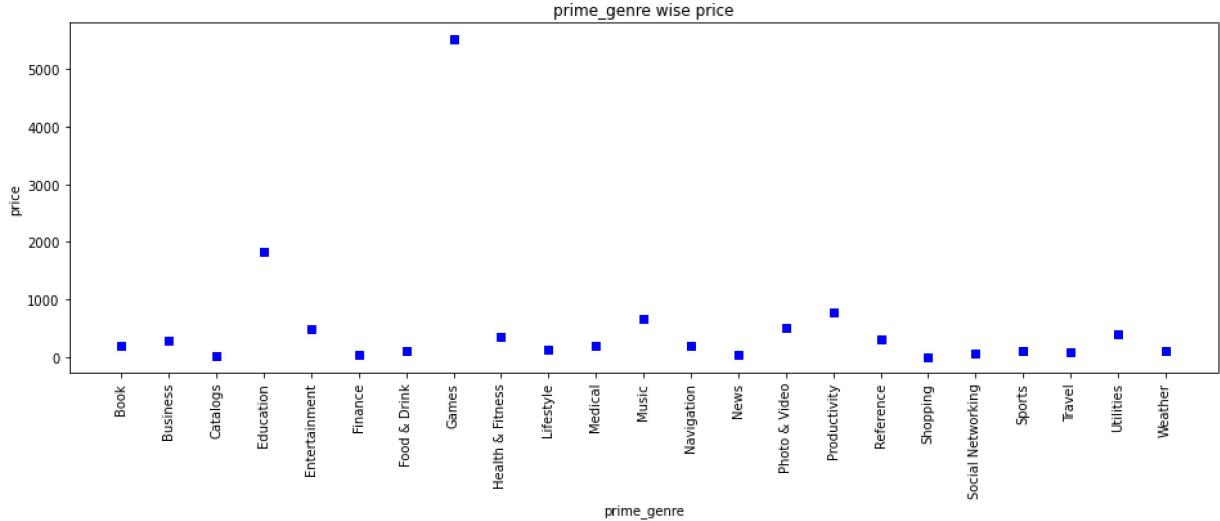
```
In [63]: plt.figure(figsize=(16,5))
plt.plot(x,'ro',color='green')
plt.xticks(rotation=90)
plt.title('prime_genre wise user_rating')
plt.xlabel('prime_genre')
plt.ylabel('user_rating')
plt.show()
```



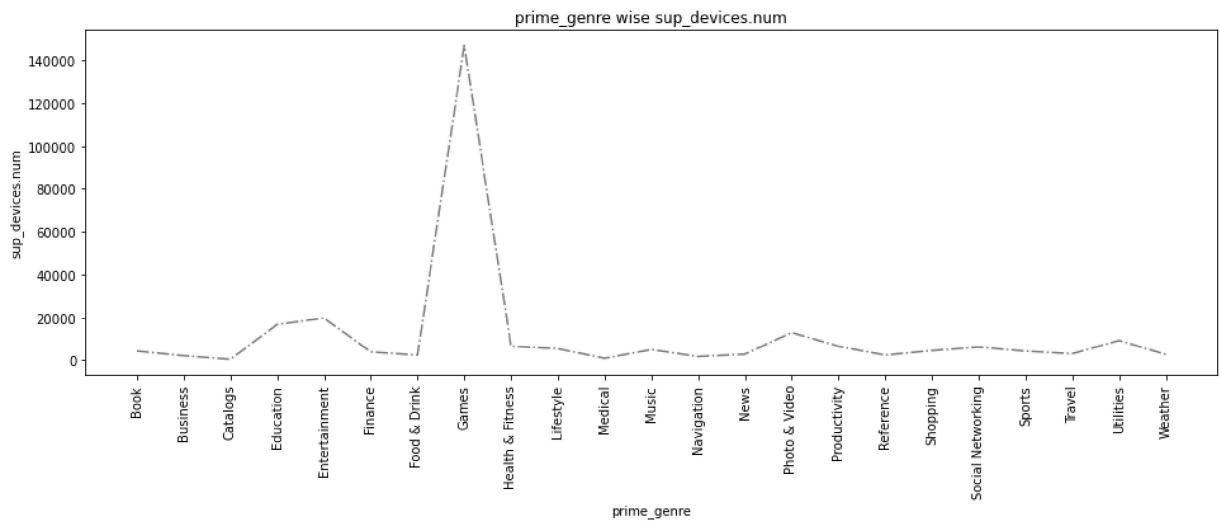
```
In [70]: plt.figure(figsize=(16,5))
plt.plot(y,'r--',color='red')
plt.xticks(rotation=90)
plt.title('prime_genre wise user_rating_ver')
plt.xlabel('prime_genre')
plt.ylabel('user_rating_ver')
plt.show()
```



```
In [71]: plt.figure(figsize=(16,5))
plt.plot(z,'bs',color='blue')
plt.xticks(rotation=90)
plt.title('prime_genre wise price')
plt.xlabel('prime_genre')
plt.ylabel('price')
plt.show()
```



```
In [72]: plt.figure(figsize=(16,5))
plt.plot(a,'-.',color='grey')
plt.xticks(rotation=90)
plt.title('prime_genre wise sup_devices.num')
plt.xlabel('prime_genre')
plt.ylabel('sup_devices.num')
plt.show()
```



```
In [ ]:
```