In [62]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [63]:

```
data = pd.read_csv("flipkart_smartphones.csv")
```

In [64]:

```
data.head()
```

Out[64]:

	title	ram	brand	url	product_id	
0	POCO C31 (Royal Blue, 64 GB)	4 GB RAM	POCO	https://www.flipkart.com/poco-c31-royal-blue-6...	MOBG73E7GKQK4KZP	LSTM
1	POCO C31 (Shadow Gray, 64 GB)	4 GB RAM	POCO	https://www.flipkart.com/poco-c31-shadow-gray-...	MOBG73E7UBFXXMCH	LSTM
2	realme C35 (Glowing Green, 64 GB)	4 GB RAM	realme	https://www.flipkart.com/realme-c35-glowing-gr...	MOBGBTHFSKHF8RAU	LSTMC
3	OPPO K10 (Black Carbon, 128 GB)	6 GB RAM	OPPO	https://www.flipkart.com/oppo-k10-black-carbon...	MOBGCFUHMDFSCM9W	LSTMC
4	MOTOROLA G60 (Soft Silver, 128 GB)	6 GB RAM	MOTOROLA	https://www.flipkart.com/motorola-g60-soft-sil...	MOBG9CJ6G5GCFAH4	LSTM

In [65]:

```
data.tail()
```

Out[65]:

	title	ram	brand	url	product_id	
842	OPPO A37f (Black, 16 GB)	2 GB RAM	OPPO	https://www.flipkart.com/oppo-a37f-black-16-gb...	MOBEWMAUFU9AFWUH	LS
843	KARBONN K9 Kavach (Champagne, 16 GB)	2 GB RAM	KARBONN	https://www.flipkart.com/karbonn-k9-kavach-cha...	MOBFYJC5CM5MHRMJ	LS
844	OPPO Reno2 (Ocean Blue, 256 GB)	8 GB RAM	OPPO	https://www.flipkart.com/oppo-reno2-ocean-blue...	MOBFJY8YRNGGZPVD	LS
845	SAMSUNG Galaxy A53 (Light Blue, 128 GB)	8 GB RAM	SAMSUNG	https://www.flipkart.com/samsung-galaxy-a53-li...	MOBGCFVYUHHUJFNY	L
846	YU Ace (Rose Gold, 16 GB)	2 GB RAM	YU	https://www.flipkart.com/yu-ace-rose-gold-16-g...	MOBF8HV9DHJJYXGH	L

In [66]:

```
data.shape
```

Out[66]:

```
(847, 19)
```

In [67]:

```
data.columns
```

Out[67]:

```
Index(['title', 'ram', 'brand', 'url', 'product_id', 'listing_id',
       'highlights', 'availability', 'selling_price', 'original_price',
       'currency', 'avg_rating', 'ratings_count', 'reviews_count',
       'one_stars_count', 'two_stars_count', 'three_stars_count',
       'four_stars_count', 'five_stars_count'],
      dtype='object')
```

In [68]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 847 entries, 0 to 846
Data columns (total 19 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   title              847 non-null    object 
 1   ram                746 non-null    object 
 2   brand              847 non-null    object 
 3   url                847 non-null    object 
 4   product_id         847 non-null    object 
 5   listing_id         847 non-null    object 
 6   highlights         847 non-null    object 
 7   availability       847 non-null    object 
 8   selling_price      847 non-null    int64  
 9   original_price     847 non-null    int64  
 10  currency           847 non-null    object 
 11  avg_rating         847 non-null    float64
 12  ratings_count      847 non-null    int64  
 13  reviews_count      847 non-null    int64  
 14  one_stars_count    847 non-null    int64  
 15  two_stars_count    847 non-null    int64  
 16  three_stars_count  847 non-null    int64  
 17  four_stars_count   847 non-null    int64  
 18  five_stars_count   847 non-null    int64  
dtypes: float64(1), int64(9), object(9)
memory usage: 125.9+ KB
```

In [69]:

```
data.describe()
```

Out[69]:

	selling_price	original_price	avg_rating	ratings_count	reviews_count	one_stars_co
count	847.000000	847.000000	847.000000	847.000000	847.000000	847.000(
mean	24810.768595	27748.353011	4.178749	37161.184179	2989.429752	1866.4958
std	27771.028820	28466.642669	0.723661	85815.344152	6679.098769	4000.221(
min	3780.000000	4999.000000	0.000000	0.000000	0.000000	0.000(
25%	10990.000000	12999.000000	4.200000	874.500000	70.000000	63.000(
50%	15499.000000	17999.000000	4.300000	4237.000000	441.000000	331.000(
75%	24999.000000	27999.000000	4.400000	31605.000000	2910.500000	1848.000(
max	179900.000000	179900.000000	5.000000	912314.000000	71867.000000	36443.000(

In [70]:

```
data.isnull().sum()
```

Out[70]:

```
title                  0
ram                  101
brand                  0
url                    0
product_id             0
listing_id             0
highlights             0
availability           0
selling_price          0
original_price         0
currency               0
avg_rating             0
ratings_count          0
reviews_count          0
one_stars_count        0
two_stars_count        0
three_stars_count      0
four_stars_count       0
five_stars_count       0
dtype: int64
```

In [71]:

```
data.dropna(inplace=True)
```

In [72]:

```
data.isnull().any().any()
```

Out[72]:

```
False
```

In [73]:

```
data.nunique()
```

Out[73]:

```
title                651
ram                    7
brand                 26
url                  746
product_id           746
listing_id           746
highlights           453
availability           4
selling_price        301
original_price       126
currency               1
avg_rating            19
ratings_count        374
reviews_count        333
one_stars_count      323
two_stars_count      283
three_stars_count    325
four_stars_count     352
five_stars_count     377
dtype: int64
```

In [74]:

```
data['brand'].unique()
```

Out[74]:

```
array(['POCO', 'realme', 'OPPO', 'MOTOROLA', 'REDMI', 'SAMSUNG', 'vivo',
       'Infinix', 'Mi', 'GIONEE', 'Itel', 'Micromax', 'Tecno',
       'MarQ By Flipkart', 'LAVA', 'Nokia', 'TCL', 'ASUS', 'Panasonic',
       'Alcatel', 'Maplin', 'YU', 'Coolpad', 'XOLO', 'Jmax', 'KARBONN'],
      dtype=object)
```

In [75]:

```
redmi_smartphone = data[data['brand'] == 'REDMI']
```

In [76]:

```
redmi_smartphone.head()
```

Out[76]:

	title	ram	brand	url	product_id	
11	REDMI 10 (Caribbean Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-caribbean-gr...	MOBGC9GYQVJHKH76	LSTMOBG
13	REDMI 10 (Midnight Black, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-midnight-bla...	MOBGC9GYX2QQXKWK	LSTMOBGC
16	REDMI 9i Sport (Carbon Black, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-carbon...	MOBG6WQWRGRRDBH6	LSTMOBG6W
35	REDMI 9i Sport (Coral Green, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-coral-...	MOBG6WQWJRFZ5WDG	LSTMOBG6V
42	REDMI Note 10 Pro (Dark Night, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-pro-dar...	MOBGB725PZQVNSUH	LSTMOBGB

In [77]:

```
redmi_smartphone.tail()
```

Out[77]:

	title	ram	brand	url	product_id	
682	REDMI Note 10 (Aqua Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-aqua-gr...	MOBGF47CCGXUZPAP	LSTMOBGF47C(
704	REDMI Note 9 (Scarlet Red, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-9-scarlet-...	MOBFUNP3DYNG7HX2	LSTMOBFUNP3I
760	REDMI Note 10T 5G (Metallic Blue, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10t-5g-met...	MOBG5GQZXZBKERSW	LSTMOBG5GQZX
793	REDMI NOTE 10 LITE (Aurora Blue, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-lite-au...	MOBG8YVAGZVXE86F	LSTMOBG8YVA(
829	REDMI Note 10S (Deep Sea Blue, 128 GB)	8 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10s-deep-s...	MOBG3YW35FNJZJGY	LSTMOBG3YW3

In [78]:

```
redmi_smartphone['title'].value_counts()
```

Out[78]:

```
REDMI Note 10S (Deep Sea Blue, 128 GB)            4
REDMI Note 10S (Frost White, 128 GB)              3
REDMI Note 10 Pro (Vintage Bronze, 128 GB)        3
REDMI Note 10 Pro (Dark Night, 128 GB)            3
REDMI Note 10S (Cosmic Purple, 128 GB)            3
REDMI Note 10S (Shadow Black, 128 GB)             2
REDMI Note 10T 5G (Graphite Black, 128 GB)        2
REDMI Note 10T 5G (Metallic Blue, 128 GB)         2
REDMI Note 10T 5G (Chromium White, 64 GB)         2
REDMI Note 10S (Frost White, 64 GB)               2
REDMI NOTE 10 LITE (Aurora Blue, 128 GB)          2
REDMI Note 10T 5G (Mint Green, 64 GB)             2
REDMI Note 10S (Cosmic Purple, 64 GB)             2
REDMI Note 11S (Polar White, 128 GB)              2
REDMI 9 Power (Blazing Blue, 128 GB)              2
REDMI Note 10 Pro (Glacial Blue, 128 GB)          2
REDMI Note 11S (Horizon Blue, 128 GB)             2
REDMI Note 10T 5G (Metallic Blue, 64 GB)          2
REDMI NOTE 10 LITE (Champagne Gold, 128 GB)       2
REDMI Note 10 Pro Max (Vintage Bronze, 128 GB)    1
REDMI 10 Prime (Phantom Black, 128 GB)            1
REDMI 9i (Midnight Black, 64 GB)                  1
REDMI 9i (Nature Green, 64 GB)                    1
REDMI 10 (Pacific Blue, 128 GB)                   1
REDMI NOTE 10 LITE (Aurora Blue, 64 GB)           1
REDMI 9 Activ (Carbon Black, 64 GB)               1
REDMI Note 10 (Aqua Green, 128 GB)                1
REDMI 10 (Midnight Black, 64 GB)                  1
REDMI Note 9 (Scarlet Red, 128 GB)                1
REDMI 9 Activ (Metallic Purple, 64 GB)            1
REDMI Note 9 Pro Max (Aurora Blue, 128 GB)        1
REDMI Note 10 (Frost White, 128 GB)               1
REDMI Note 10 Pro (Dark Nebula, 128 GB)           1
REDMI 9i (Sea Blue, 64 GB)                        1
REDMI 9 Activ (Carbon Black, 128 GB)              1
REDMI 9 Activ (Coral Green, 128 GB)               1
REDMI Note 11S (Space Black, 128 GB)              1
REDMI Note 10T 5G (Mint Green, 128 GB)            1
REDMI 10 Prime (Bifrost Blue, 128 GB)             1
REDMI 10 (Midnight Black, 128 GB)                 1
REDMI 9i Sport (Metallic Blue, 64 GB)             1
REDMI 10 (Caribbean Green, 64 GB)                 1
REDMI 10 Prime (Astral White, 64 GB)              1
REDMI Note 10T 5G (Chromium White, 128 GB)        1
REDMI 9 Activ (Coral Green, 64 GB)                1
REDMI 9 Prime (Mint Green, 64 GB)                 1
REDMI 10 Prime (Astral White, 128 GB)             1
REDMI 9 Power (Electric Green, 128 GB)            1
REDMI 10 (Pacific Blue, 64 GB)                    1
REDMI 9i Sport (Carbon Black, 64 GB)              1
REDMI Note 10S (Shadow Black, 64 GB)              1
REDMI Note 8 Pro (Halo White, 128 GB)             1
REDMI 8A Dual (Sky White, 32 GB)                  1
REDMI 9 Power (Blazing Blue, 64 GB)               1
```

```
REDMI 9i Sport (Coral Green, 64 GB)              1
REDMI 10 (Caribbean Green, 128 GB)               1
REDMI Note 10 Lite (Glacier White, 128 GB)       1
REDMI Note 9 Pro Max (Champagne Gold, 64 GB)     1
Name: title, dtype: int64
```

In [79]:

```
redmi_smartphone['title'].value_counts().sum()
```

Out[79]:

```
83
```

In [80]:

```
redmi_smartphone.index = range(len(redmi_smartphone.index))
```

In [81]:

```
redmi_smartphone.head()
```

Out[81]:

	title	ram	brand	url	product_id	
0	REDMI 10 (Caribbean Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-caribbean-gr...	MOBGC9GYQVJHKH76	LSTMOBGC
1	REDMI 10 (Midnight Black, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-midnight-bla...	MOBGC9GYX2QQXKWK	LSTMOBGC9
2	REDMI 9i Sport (Carbon Black, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-carbon...	MOBG6WQWRGRRDBH6	LSTMOBG6WQ
3	REDMI 9i Sport (Coral Green, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-coral-...	MOBG6WQWJRFZ5WDG	LSTMOBG6W
4	REDMI Note 10 Pro (Dark Night, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-pro-dar...	MOBGB725PZQVNSUH	LSTMOBGB7

In [82]:

```
redmi_smartphone.tail()
```

Out[82]:

	title	ram	brand	url	product_id	
78	REDMI Note 10 (Aqua Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-aqua-gr...	MOBGF47CCGXUZPAP	LSTMOBGF47CC(
79	REDMI Note 9 (Scarlet Red, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-9-scarlet-...	MOBFUNP3DYNG7HX2	LSTMOBFUNP3D'
80	REDMI Note 10T 5G (Metallic Blue, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10t-5g-met...	MOBG5GQZXZBKERSW	LSTMOBG5GQZXZ
81	REDMI NOTE 10 LITE (Aurora Blue, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-lite-au...	MOBG8YVAGZVXE86F	LSTMOBG8YVAG
82	REDMI Note 10S (Deep Sea Blue, 128 GB)	8 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10s-deep-s...	MOBG3YW35FNJZJGY	LSTMOBG3YW35

In [83]:

```
new = redmi_smartphone["ram"].str.split(" ", n = 2, expand = True)
```

In [84]:

```
redmi_smartphone['ram_val'] = new[0]
```

```
<ipython-input-84-c302d89de866>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  redmi_smartphone['ram_val'] = new[0]
```

In [85]:

```
redmi_smartphone['ram_val'].head()
```

Out[85]:

```
0    6
1    6
2    4
3    4
4    6
Name: ram_val, dtype: object
```

In [86]:

```
redmi_smartphone['ram_val'] = redmi_smartphone['ram_val'].astype(int)
```

```
<ipython-input-86-01a8a7b5ec86>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  redmi_smartphone['ram_val'] = redmi_smartphone['ram_val'].astype(int)
```

In [87]:

```
redmi_smartphone.head()
```

Out[87]:

	title	ram	brand	url	product_id	
0	REDMI 10 (Caribbean Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-caribbean-gr...	MOBGC9GYQVJHKH76	LSTMOBGC
1	REDMI 10 (Midnight Black, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-10-midnight-bla...	MOBGC9GYX2QQXKWK	LSTMOBGC9
2	REDMI 9i Sport (Carbon Black, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-carbon...	MOBG6WQWRGRRDBH6	LSTMOBG6WQ
3	REDMI 9i Sport (Coral Green, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-9i-sport-coral-...	MOBG6WQWJRFZ5WDG	LSTMOBG6W
4	REDMI Note 10 Pro (Dark Night, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-pro-dar...	MOBGB725PZQVNSUH	LSTMOBGB7

In [88]:

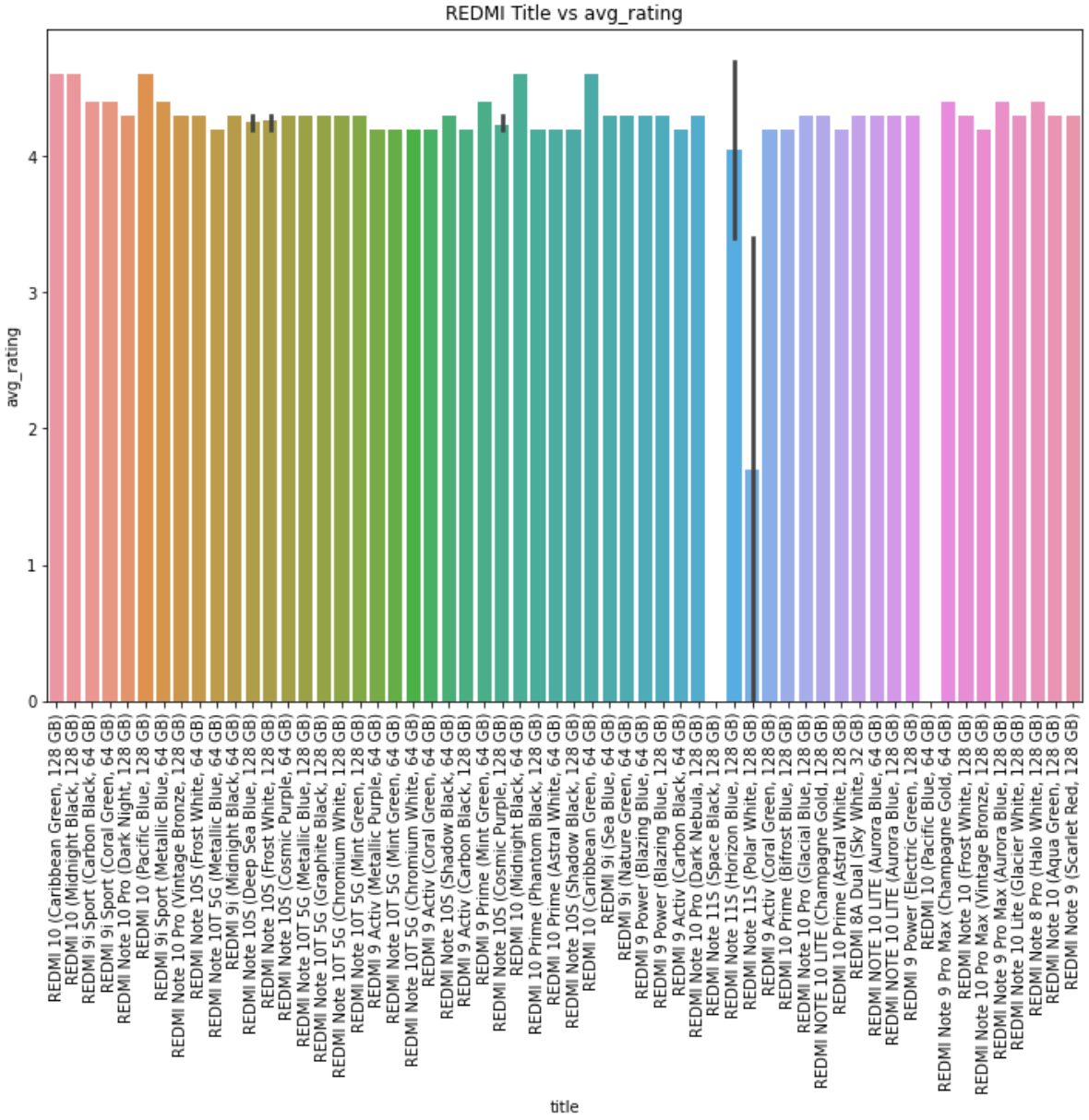
```
redmi_smartphone.tail()
```

Out[88]:

	title	ram	brand	url	product_id	
78	REDMI Note 10 (Aqua Green, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-aqua-gr...	MOBGF47CCGXUZPAP	LSTMOBGF47CC
79	REDMI Note 9 (Scarlet Red, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-9-scarlet-...	MOBFUNP3DYNG7HX2	LSTMOBFUNP3D
80	REDMI Note 10T 5G (Metallic Blue, 64 GB)	4 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10t-5g-met...	MOBG5GQZXZBKERSW	LSTMOBG5GQZXZ
81	REDMI NOTE 10 LITE (Aurora Blue, 128 GB)	6 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10-lite-au...	MOBG8YVAGZVXE86F	LSTMOBG8YVAG
82	REDMI Note 10S (Deep Sea Blue, 128 GB)	8 GB RAM	REDMI	https://www.flipkart.com/redmi-note-10s-deep-s...	MOBG3YW35FNJZJGY	LSTMOBG3YW35

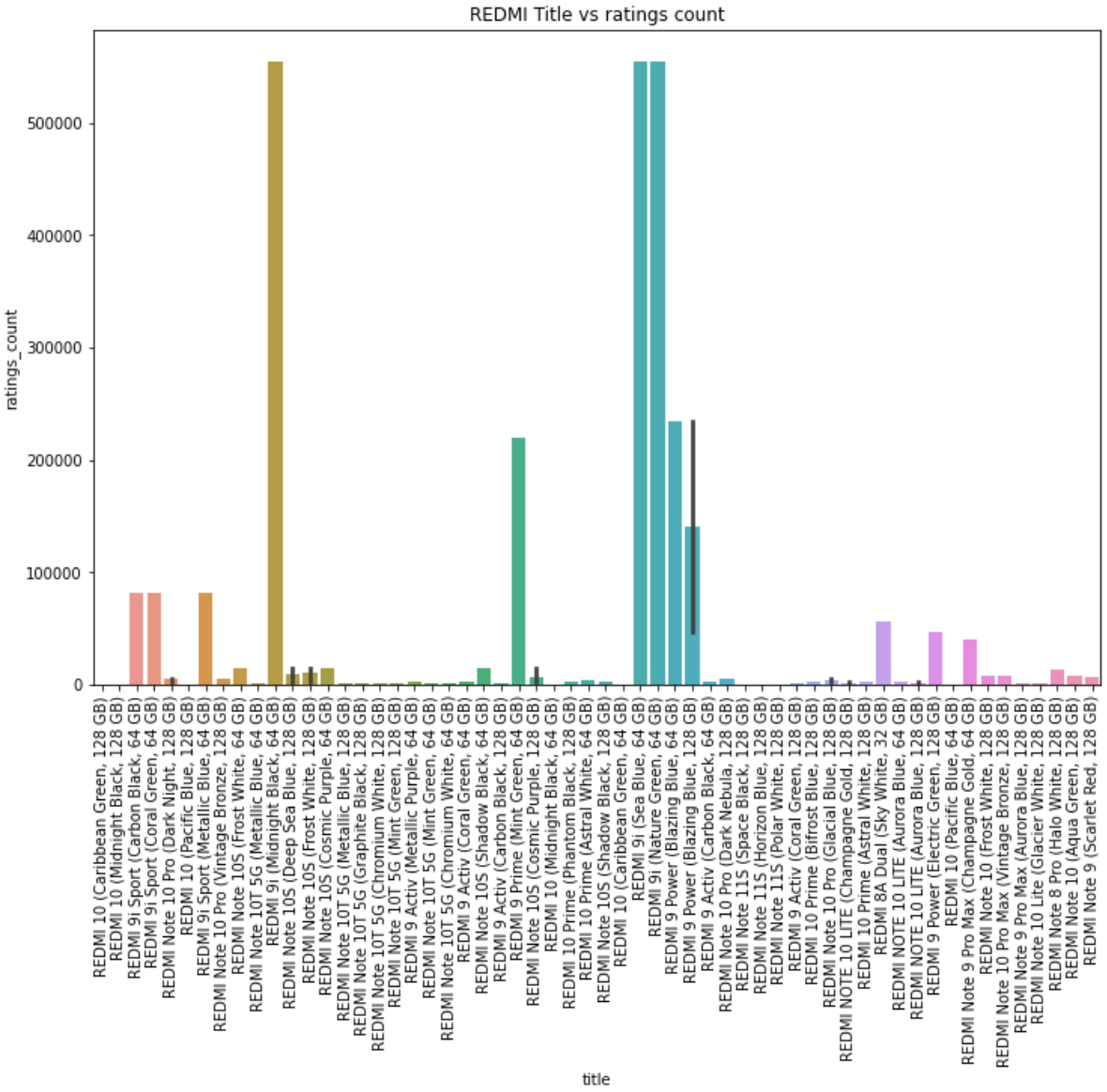
In [89]:

```
plt.figure(figsize = (12,8))
plt.xticks(rotation = 90)
sns.barplot(x = redmi_smartphone.title, y = redmi_smartphone.avg_rating)
plt.title("REDMI Title vs avg_rating")
plt.show()
```



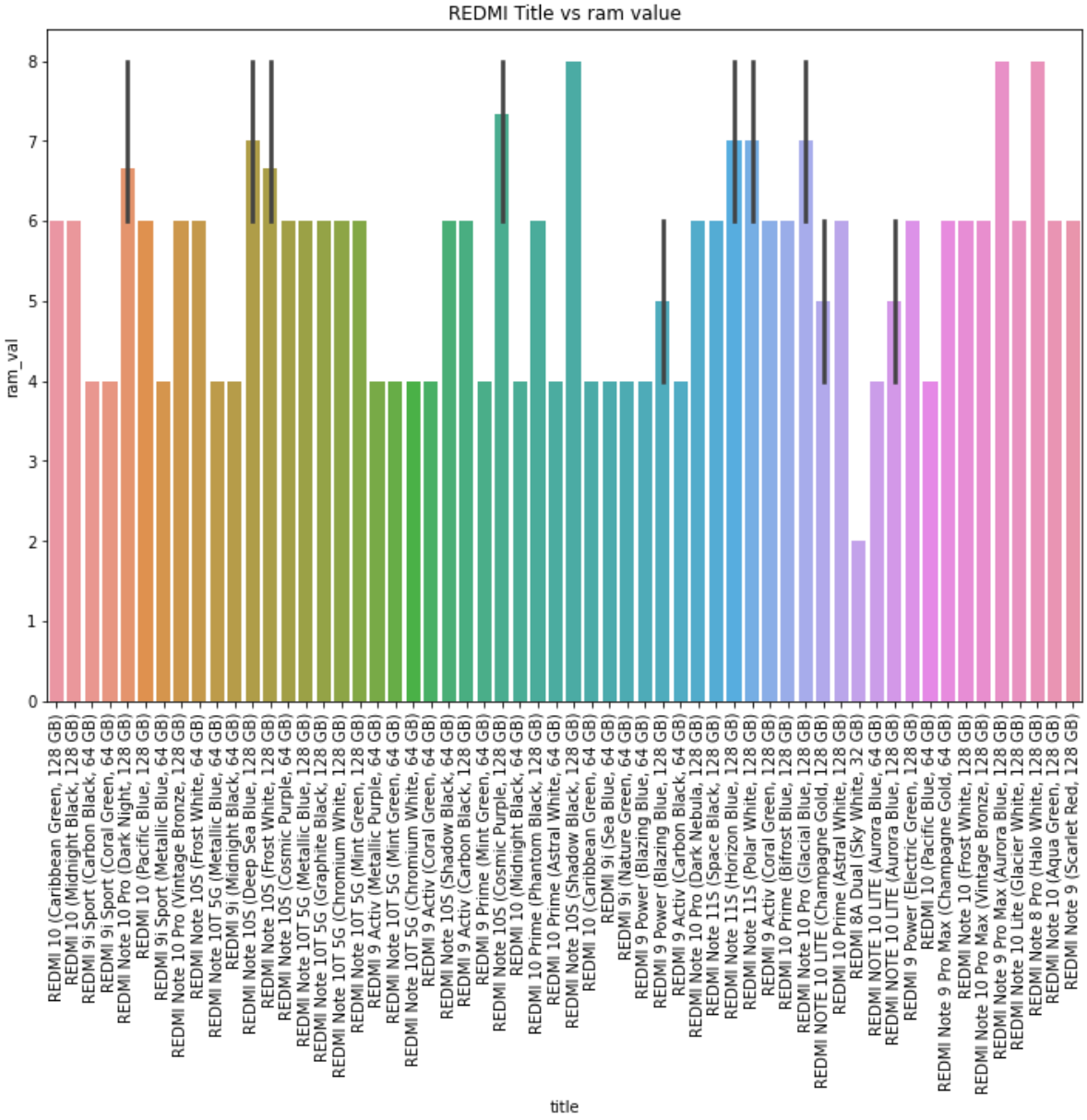
In [90]:

```
plt.figure(figsize = (12,8))
plt.xticks(rotation = 90)
sns.barplot(x = redmi_smartphone.title, y = redmi_smartphone.ratings_count)
plt.title("REDMI Title vs ratings count")
plt.show()
```



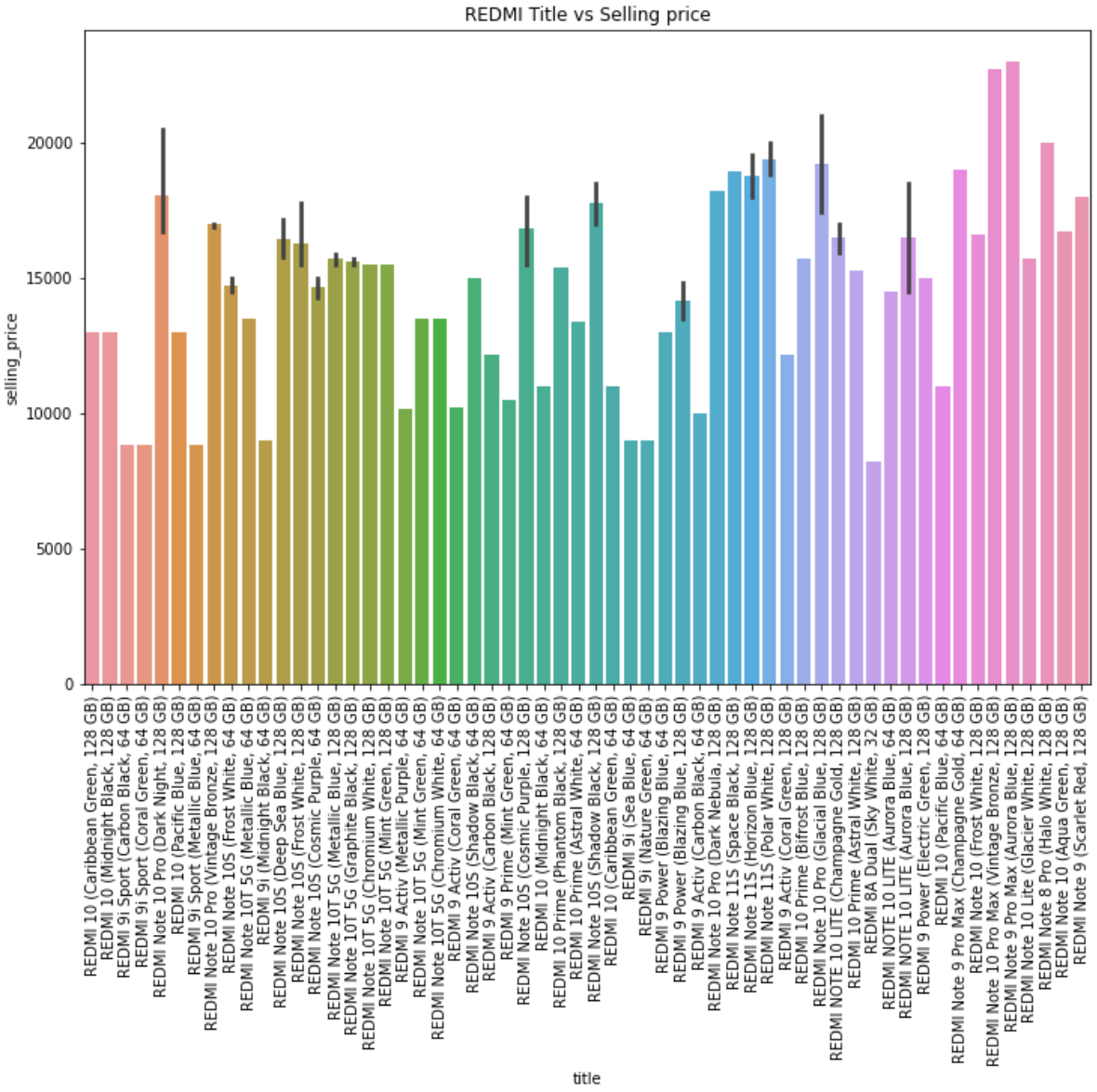
In [97]:

```
plt.figure(figsize = (12,8))
plt.xticks(rotation = 90)
sns.barplot(x = redmi_smartphone.title, y = redmi_smartphone.ram_val)
plt.title("REDMI Title vs ram value")
plt.show()
```



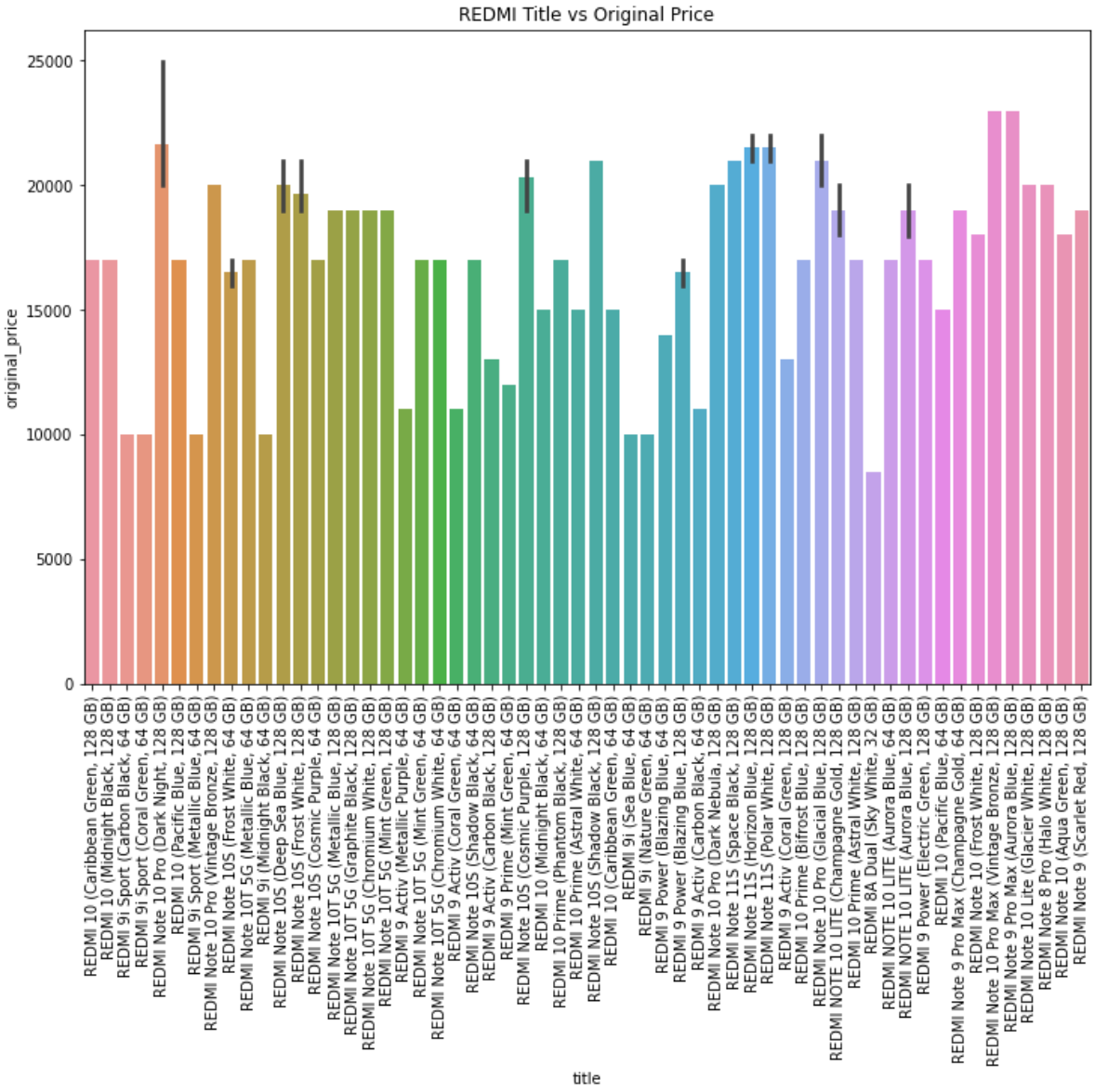
In [98]:

```
plt.figure(figsize = (12,8))
plt.xticks(rotation = 90)
sns.barplot(x = redmi_smartphone.title, y = redmi_smartphone.selling_price)
plt.title("REDMI Title vs Selling price")
plt.show()
```



In [99]:

```
plt.figure(figsize = (12,8))
plt.xticks(rotation = 90)
sns.barplot(x = redmi_smartphone.title, y = redmi_smartphone.original_price)
plt.title("REDMI Title vs Original Price")
plt.show()
```



In [100]:

```
x = redmi_smartphone.drop(['title', 'ram', 'brand', 'url', 'product_id', 'listing_id',
                           'availability', 'selling_price', 'currency'], axis = 1)
y = redmi_smartphone['selling_price']
```

In [101]:

```
x.shape
```

Out[101]:

```
(83, 10)
```

In [102]:

```
y.shape
```

Out[102]:

```
(83,)
```

In [103]:

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)
```

In [104]:

```
model_india=LinearRegression()
model_india.fit(x,y)
```

Out[104]:

```
LinearRegression()
```

In [105]:

```
print("Training Accuracy :", model_india.score(X_train, y_train))
print("Testing Accuracy :", model_india.score(X_test, y_test))
```

```
Training Accuracy : 0.9132131428738655
Testing Accuracy : 0.795688677544443
```

In [106]:

```
print("Coefficient: ",model_india.coef_)
print("intercept: ",model_india.intercept_)
pre = model_india.predict(x)
```

```
Coefficient:  [ 8.42315002e-01 -8.29447236e+01  4.15201906e-01  2.19273271
e+00
 -3.83308885e+00  6.63225275e+00 -5.94854621e-01 -1.56023499e+00
 -2.28872376e-01  2.56855281e+02]
intercept:  -931.4982076256183
```

In [108]:

```
from sklearn.metrics import mean_squared_error
print("By function: ",mean_squared_error(y,model_india.predict(x)))
```

```
By function:  1050679.4536276404
```