akash-project-13

April 28, 2023

1 Google App Store Data Analysis

Project Title : Google App Store Data Analysis

Technologies : Data Science

Domain : Technology

Project Difficulties level : Intermediate

1.0.1 Author : AKASH.V

2 Import Libraries

The following libraries will be used: - 1. `numpy`: numerical analysis - 2. `pandas`: data analysis - 3. `seaborn and matplotlib`: data vizualisation - 4. `opendatasets`: to download dataset

[ ]:

```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import opendatasets as od
import seaborn as sns
import matplotlib.style as sty
import matplotlib.ticker as mtick
import matplotlib.gridspec as grid_specTrue
```

Read the dataset using pandas

[3]:

```python
df = pd.read_csv('Google Apps data.csv.csv')
```

Let us view the dataset and see the content

[4]:

```python
df.head()
```

[4]:

```
                                                 App        Category  Rating  \
0     Photo Editor & Candy Camera & Grid & ScrapBook  ART_AND_DESIGN     4.1
1                                Coloring book moana  ART_AND_DESIGN     3.9
2  U Launcher Lite – FREE Live Cool Themes, Hide …  ART_AND_DESIGN     4.7
3                              Sketch - Draw & Paint  ART_AND_DESIGN     4.5
4              Pixel Draw - Number Art Coloring Book  ART_AND_DESIGN     4.3
```

```
  Reviews  Size     Installs  Type Price Content Rating  \
0     159   19M      10,000+  Free     0       Everyone
1     967   14M     500,000+  Free     0       Everyone
2   87510  8.7M   5,000,000+  Free     0       Everyone
3  215644   25M  50,000,000+  Free     0           Teen
4     967  2.8M     100,000+  Free     0       Everyone

                      Genres      Last Updated         Current Ver  \
0               Art & Design   January 7, 2018               1.0.0
1  Art & Design;Pretend Play  January 15, 2018               2.0.0
2               Art & Design    August 1, 2018               1.2.4
3               Art & Design      June 8, 2018  Varies with device
4    Art & Design;Creativity     June 20, 2018                 1.1

    Android Ver
0  4.0.3 and up
1  4.0.3 and up
2  4.0.3 and up
3    4.2 and up
4    4.4 and up
```

Overview of the dataset

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   App             10841 non-null  object
 1   Category        10841 non-null  object
 2   Rating          9367 non-null   float64
 3   Reviews         10841 non-null  object
 4   Size            10841 non-null  object
 5   Installs        10841 non-null  object
 6   Type            10840 non-null  object
 7   Price           10841 non-null  object
 8   Content Rating  10840 non-null  object
 9   Genres          10841 non-null  object
 10  Last Updated    10841 non-null  object
 11  Current Ver     10833 non-null  object
 12  Android Ver     10838 non-null  object
dtypes: float64(1), object(12)
memory usage: 1.1+ MB
None
```

The dataset has 13 columns and 10841 rows

Statistical Summary

[6]:

```
df.describe().transpose()
```

[6]:

```
         count      mean       std  min  25%  50%  75%   max
Rating  9367.0  4.193338  0.537431  1.0  4.0  4.3  4.5  19.0
```

As it stands, only Ratings column as numerical values, others are object

[7]:

```
# name of columns in the dataset

df.columns
```

[7]:

```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
       'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
       'Android Ver'],
      dtype='object')
```

3 Data Cleaning

- ___I will replace the whitespace(s) in the columns name as '_' for better understanding___

[8]:

```
df.columns = df.columns.str.replace(' ', '_')
```

Let us check if the changes occur or not

[9]:

```
df.columns
```

[9]:

```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
       'Price', 'Content_Rating', 'Genres', 'Last_Updated', 'Current_Ver',
       'Android_Ver'],
      dtype='object')
```

- **Next, I will check the unique values in each column**

[10]:

```
for col in df:
    print(col)
    print(df[col].unique())
    print('-'*20)
```

```
App
['Photo Editor & Candy Camera & Grid & ScrapBook' 'Coloring book moana'
 'U Launcher Lite – FREE Live Cool Themes, Hide Apps' …
 'Parkinson Exercices FR' 'The SCP Foundation DB fr nn5n'
 'iHoroscope - 2018 Daily Horoscope & Astrology']
--------------------
Category
```

```
['ART_AND_DESIGN' 'AUTO_AND_VEHICLES' 'BEAUTY' 'BOOKS_AND_REFERENCE'
 'BUSINESS' 'COMICS' 'COMMUNICATION' 'DATING' 'EDUCATION' 'ENTERTAINMENT'
 'EVENTS' 'FINANCE' 'FOOD_AND_DRINK' 'HEALTH_AND_FITNESS' 'HOUSE_AND_HOME'
 'LIBRARIES_AND_DEMO' 'LIFESTYLE' 'GAME' 'FAMILY' 'MEDICAL' 'SOCIAL'
 'SHOPPING' 'PHOTOGRAPHY' 'SPORTS' 'TRAVEL_AND_LOCAL' 'TOOLS'
 'PERSONALIZATION' 'PRODUCTIVITY' 'PARENTING' 'WEATHER' 'VIDEO_PLAYERS'
 'NEWS_AND_MAGAZINES' 'MAPS_AND_NAVIGATION' '1.9']
--------------------
Rating
[ 4.1  3.9  4.7  4.5  4.3  4.4  3.8  4.2  4.6  3.2  4.   nan  4.8  4.9
  3.6  3.7  3.3  3.4  3.5  3.1  5.   2.6  3.   1.9  2.5  2.8  2.7  1.
  2.9  2.3  2.2  1.7  2.   1.8  2.4  1.6  2.1  1.4  1.5  1.2 19. ]
--------------------
Reviews
['159' '967' '87510' … '603' '1195' '398307']
--------------------
Size
['19M' '14M' '8.7M' '25M' '2.8M' '5.6M' '29M' '33M' '3.1M' '28M' '12M'
 '20M' '21M' '37M' '2.7M' '5.5M' '17M' '39M' '31M' '4.2M' '7.0M' '23M'
 '6.0M' '6.1M' '4.6M' '9.2M' '5.2M' '11M' '24M' 'Varies with device'
 '9.4M' '15M' '10M' '1.2M' '26M' '8.0M' '7.9M' '56M' '57M' '35M' '54M'
 '201k' '3.6M' '5.7M' '8.6M' '2.4M' '27M' '2.5M' '16M' '3.4M' '8.9M'
 '3.9M' '2.9M' '38M' '32M' '5.4M' '18M' '1.1M' '2.2M' '4.5M' '9.8M' '52M'
 '9.0M' '6.7M' '30M' '2.6M' '7.1M' '3.7M' '22M' '7.4M' '6.4M' '3.2M'
 '8.2M' '9.9M' '4.9M' '9.5M' '5.0M' '5.9M' '13M' '73M' '6.8M' '3.5M'
 '4.0M' '2.3M' '7.2M' '2.1M' '42M' '7.3M' '9.1M' '55M' '23k' '6.5M' '1.5M'
 '7.5M' '51M' '41M' '48M' '8.5M' '46M' '8.3M' '4.3M' '4.7M' '3.3M' '40M'
 '7.8M' '8.8M' '6.6M' '5.1M' '61M' '66M' '79k' '8.4M' '118k' '44M' '695k'
 '1.6M' '6.2M' '18k' '53M' '1.4M' '3.0M' '5.8M' '3.8M' '9.6M' '45M' '63M'
 '49M' '77M' '4.4M' '4.8M' '70M' '6.9M' '9.3M' '10.0M' '8.1M' '36M' '84M'
 '97M' '2.0M' '1.9M' '1.8M' '5.3M' '47M' '556k' '526k' '76M' '7.6M' '59M'
 '9.7M' '78M' '72M' '43M' '7.7M' '6.3M' '334k' '34M' '93M' '65M' '79M'
 '100M' '58M' '50M' '68M' '64M' '67M' '60M' '94M' '232k' '99M' '624k'
 '95M' '8.5k' '41k' '292k' '11k' '80M' '1.7M' '74M' '62M' '69M' '75M'
 '98M' '85M' '82M' '96M' '87M' '71M' '86M' '91M' '81M' '92M' '83M' '88M'
 '704k' '862k' '899k' '378k' '266k' '375k' '1.3M' '975k' '980k' '4.1M'
 '89M' '696k' '544k' '525k' '920k' '779k' '853k' '720k' '713k' '772k'
 '318k' '58k' '241k' '196k' '857k' '51k' '953k' '865k' '251k' '930k'
 '540k' '313k' '746k' '203k' '26k' '314k' '239k' '371k' '220k' '730k'
 '756k' '91k' '293k' '17k' '74k' '14k' '317k' '78k' '924k' '902k' '818k'
 '81k' '939k' '169k' '45k' '475k' '965k' '90M' '545k' '61k' '283k' '655k'
 '714k' '93k' '872k' '121k' '322k' '1.0M' '976k' '172k' '238k' '549k'
 '206k' '954k' '444k' '717k' '210k' '609k' '308k' '705k' '306k' '904k'
 '473k' '175k' '350k' '383k' '454k' '421k' '70k' '812k' '442k' '842k'
 '417k' '412k' '459k' '478k' '335k' '782k' '721k' '430k' '429k' '192k'
 '200k' '460k' '728k' '496k' '816k' '414k' '506k' '887k' '613k' '243k'
 '569k' '778k' '683k' '592k' '319k' '186k' '840k' '647k' '191k' '373k'
 '437k' '598k' '716k' '585k' '982k' '222k' '219k' '55k' '948k' '323k'
 '691k' '511k' '951k' '963k' '25k' '554k' '351k' '27k' '82k' '208k' '913k'
 '514k' '551k' '29k' '103k' '898k' '743k' '116k' '153k' '209k' '353k'
 '499k' '173k' '597k' '809k' '122k' '411k' '400k' '801k' '787k' '237k'
 '50k' '643k' '986k' '97k' '516k' '837k' '780k' '961k' '269k' '20k' '498k'
 '600k' '749k' '642k' '881k' '72k' '656k' '601k' '221k' '228k' '108k'
 '940k' '176k' '33k' '663k' '34k' '942k' '259k' '164k' '458k' '245k'
 '629k' '28k' '288k' '775k' '785k' '636k' '916k' '994k' '309k' '485k'
 '914k' '903k' '608k' '500k' '54k' '562k' '847k' '957k' '688k' '811k'
 '270k' '48k' '329k' '523k' '921k' '874k' '981k' '784k' '280k' '24k'
 '518k' '754k' '892k' '154k' '860k' '364k' '387k' '626k' '161k' '879k'
 '39k' '970k' '170k' '141k' '160k' '144k' '143k' '190k' '376k' '193k'
 '246k' '73k' '658k' '992k' '253k' '420k' '404k' '1,000+' '470k' '226k'
 '240k' '89k' '234k' '257k' '861k' '467k' '157k' '44k' '676k' '67k' '552k'
 '885k' '1020k' '582k' '619k']
--------------------
Installs
['10,000+' '500,000+' '5,000,000+' '50,000,000+' '100,000+' '50,000+'
 '1,000,000+' '10,000,000+' '5,000+' '100,000,000+' '1,000,000,000+'
 '1,000+' '500,000,000+' '50+' '100+' '500+' '10+' '1+' '5+' '0+' '0'
 'Free']
--------------------
Type
['Free' 'Paid' nan '0']
--------------------
Price
['0' '$4.99' '$3.99' '$6.99' '$1.49' '$2.99' '$7.99' '$5.99' '$3.49'
 '$1.99' '$9.99' '$7.49' '$0.99' '$9.00' '$5.49' '$10.00' '$24.99'
 '$11.99' '$79.99' '$16.99' '$14.99' '$1.00' '$29.99' '$12.99' '$2.49'
 '$10.99' '$1.50' '$19.99' '$15.99' '$33.99' '$74.99' '$39.99' '$3.95'
 '$4.49' '$1.70' '$8.99' '$2.00' '$3.88' '$25.99' '$399.99' '$17.99'
 '$400.00' '$3.02' '$1.76' '$4.84' '$4.77' '$1.61' '$2.50' '$1.59' '$6.49'
 '$1.29' '$5.00' '$13.99' '$299.99' '$379.99' '$37.99' '$18.99' '$389.99'
 '$19.90' '$8.49' '$1.75' '$14.00' '$4.85' '$46.99' '$109.99' '$154.99'
 '$3.08' '$2.59' '$4.80' '$1.96' '$19.40' '$3.90' '$4.59' '$15.46' '$3.04'
 '$4.29' '$2.60' '$3.28' '$4.60' '$28.99' '$2.95' '$2.90' '$1.97'
 '$200.00' '$89.99' '$2.56' '$30.99' '$3.61' '$394.99' '$1.26' 'Everyone'
 '$1.20' '$1.04']
--------------------
Content_Rating
['Everyone' 'Teen' 'Everyone 10+' 'Mature 17+' 'Adults only 18+' 'Unrated'
 nan]
--------------------
Genres
['Art & Design' 'Art & Design;Pretend Play' 'Art & Design;Creativity'
 'Art & Design;Action & Adventure' 'Auto & Vehicles' 'Beauty'
 'Books & Reference' 'Business' 'Comics' 'Comics;Creativity'
 'Communication' 'Dating' 'Education;Education' 'Education'
 'Education;Creativity' 'Education;Music & Video'
 'Education;Action & Adventure' 'Education;Pretend Play'
 'Education;Brain Games' 'Entertainment' 'Entertainment;Music & Video'
 'Entertainment;Brain Games' 'Entertainment;Creativity' 'Events' 'Finance'
 'Food & Drink' 'Health & Fitness' 'House & Home' 'Libraries & Demo'
 'Lifestyle' 'Lifestyle;Pretend Play' 'Adventure;Action & Adventure'
 'Arcade' 'Casual' 'Card' 'Casual;Pretend Play' 'Action' 'Strategy'
 'Puzzle' 'Sports' 'Music' 'Word' 'Racing' 'Casual;Creativity'
 'Casual;Action & Adventure' 'Simulation' 'Adventure' 'Board' 'Trivia'
 'Role Playing' 'Simulation;Education' 'Action;Action & Adventure'
 'Casual;Brain Games' 'Simulation;Action & Adventure'
 'Educational;Creativity' 'Puzzle;Brain Games' 'Educational;Education'
 'Card;Brain Games' 'Educational;Brain Games' 'Educational;Pretend Play'
 'Entertainment;Education' 'Casual;Education' 'Music;Music & Video'
 'Racing;Action & Adventure' 'Arcade;Pretend Play'
 'Role Playing;Action & Adventure' 'Simulation;Pretend Play'
 'Puzzle;Creativity' 'Sports;Action & Adventure'
 'Educational;Action & Adventure' 'Arcade;Action & Adventure'
 'Entertainment;Action & Adventure' 'Puzzle;Action & Adventure'
 'Strategy;Action & Adventure' 'Music & Audio;Music & Video'
 'Health & Fitness;Education' 'Adventure;Education' 'Board;Brain Games'
 'Board;Action & Adventure' 'Board;Pretend Play' 'Casual;Music & Video'
 'Role Playing;Pretend Play' 'Entertainment;Pretend Play'
 'Video Players & Editors;Creativity' 'Card;Action & Adventure' 'Medical'
 'Social' 'Shopping' 'Photography' 'Travel & Local'
 'Travel & Local;Action & Adventure' 'Tools' 'Tools;Education'
 'Personalization' 'Productivity' 'Parenting' 'Parenting;Music & Video'
 'Parenting;Education' 'Parenting;Brain Games' 'Weather'
 'Video Players & Editors' 'Video Players & Editors;Music & Video'
 'News & Magazines' 'Maps & Navigation'
 'Health & Fitness;Action & Adventure' 'Educational' 'Casino'
 'Adventure;Brain Games' 'Trivia;Education' 'Lifestyle;Education'
 'Books & Reference;Creativity' 'Books & Reference;Education'
 'Puzzle;Education' 'Role Playing;Education' 'Role Playing;Brain Games'
 'Strategy;Education' 'Racing;Pretend Play' 'Communication;Creativity'
 'February 11, 2018' 'Strategy;Creativity']
--------------------
Last_Updated
['January 7, 2018' 'January 15, 2018' 'August 1, 2018' …
 'January 20, 2014' 'February 16, 2014' 'March 23, 2014']
--------------------
Current_Ver
['1.0.0' '2.0.0' '1.2.4' … '1.0.612928' '0.3.4' '2.0.148.0']
--------------------
Android_Ver
['4.0.3 and up' '4.2 and up' '4.4 and up' '2.3 and up' '3.0 and up'
 '4.1 and up' '4.0 and up' '2.3.3 and up' 'Varies with device'
 '2.2 and up' '5.0 and up' '6.0 and up' '1.6 and up' '1.5 and up'
 '2.1 and up' '7.0 and up' '5.1 and up' '4.3 and up' '4.0.3 - 7.1.1'
 '2.0 and up' '3.2 and up' '4.4W and up' '7.1 and up' '7.0 - 7.1.1'
 '8.0 and up' '5.0 - 8.0' '3.1 and up' '2.0.1 and up' '4.1 - 7.1.1' nan
 '5.0 - 6.0' '1.0 and up' '2.2 - 7.1.1' '5.0 - 7.1.1']
--------------------
```

Check for duplicated data

[11]:
```
df[df.duplicated()]
```

[11]:
```
                                                App      Category  Rating  \
229                          Quick PDF Scanner + OCR FREE      BUSINESS     4.2
236                                                 Box      BUSINESS     4.2
239                                  Google My Business      BUSINESS     4.4
256                                 ZOOM Cloud Meetings      BUSINESS     4.4
261                            join.me - Simple Meetings      BUSINESS     4.0
…                                                     …           …       …
8643                       Wunderlist: To-Do List & Tasks  PRODUCTIVITY     4.6
8654   TickTick: To Do List with Reminder, Day Planner  PRODUCTIVITY     4.6
8658                            ColorNote Notepad Notes  PRODUCTIVITY     4.6
10049         Airway Ex - Intubate. Anesthetize. Train.       MEDICAL     4.3
10768                                              AAFP       MEDICAL     3.8

       Reviews                Size       Installs  Type Price Content_Rating  \
229      80805  Varies with device     5,000,000+  Free     0       Everyone
236     159872  Varies with device    10,000,000+  Free     0       Everyone
239      70991  Varies with device     5,000,000+  Free     0       Everyone
256      31614                 37M    10,000,000+  Free     0       Everyone
261       6989  Varies with device     1,000,000+  Free     0       Everyone
…          …                   …           …    …    …             …
8643    404610  Varies with device    10,000,000+  Free     0       Everyone
8654     25370  Varies with device     1,000,000+  Free     0       Everyone
8658   2401017  Varies with device   100,000,000+  Free     0       Everyone
10049      123                 86M        10,000+  Free     0       Everyone
10768       63                 24M        10,000+  Free     0       Everyone

             Genres       Last_Updated         Current_Ver         Android_Ver
229        Business  February 26, 2018  Varies with device        4.0.3 and up
236        Business      July 31, 2018  Varies with device  Varies with device
239        Business      July 24, 2018    2.19.0.204537701          4.4 and up
256        Business      July 20, 2018       4.1.28165.0716          4.0 and up
261        Business      July 16, 2018           4.3.0.508          4.4 and up
…                 …                  …                   …                   …
8643   Productivity      April 6, 2018  Varies with device  Varies with device
8654   Productivity     August 6, 2018  Varies with device  Varies with device
8658   Productivity      June 27, 2018  Varies with device  Varies with device
10049       Medical       June 1, 2018              0.6.88          5.0 and up
10768       Medical      June 22, 2018               2.3.1          5.0 and up
```

```
[483 rows x 13 columns]
```

There as 483 entries that are duplicate, then, there may be chance that the App name, and one of the other column has the same entries. Therefore, I will find duplicate entries across the column and drop any row with same entries

[12]:
```
df.drop_duplicates(['App', 'Size', 'Installs', 'Type', 'Genres'], inplace=True)
df.duplicated().sum()
```

[12]:
```
0
```

Next, I will take care of NaN entries in the dataset

[13]:
```
df.isna().sum()
```

[13]:
```
App                  0
Category             0
Rating            1464
Reviews              0
Size                 0
Installs             0
Type                 1
Price                0
Content_Rating       1
Genres               0
Last_Updated         0
Current_Ver          8
Android_Ver          3
dtype: int64
```

`Rating` column has 1464 `Nan` entries, while `Current_Ver` and `Android_Ver` has 8 and 3 respectively

[14]:
```
missing_rows = df.isnull()
```

[15]:
```
for col in missing_rows.columns.values.tolist():
    print(col)
    print(missing_rows[col].value_counts())
    print('-'*20)
```

```
App
False    9681
Name: App, dtype: int64
--------------------
Category
False    9681
Name: Category, dtype: int64
--------------------
Rating
```

```
False    8217
True     1464
Name: Rating, dtype: int64
--------------------
Reviews
False    9681
Name: Reviews, dtype: int64
--------------------
Size
False    9681
Name: Size, dtype: int64
--------------------
Installs
False    9681
Name: Installs, dtype: int64
--------------------
Type
False    9680
True        1
Name: Type, dtype: int64
--------------------
Price
False    9681
Name: Price, dtype: int64
--------------------
Content_Rating
False    9680
True        1
Name: Content_Rating, dtype: int64
--------------------
Genres
False    9681
Name: Genres, dtype: int64
--------------------
Last_Updated
False    9681
Name: Last_Updated, dtype: int64
--------------------
Current_Ver
False    9673
True        8
Name: Current_Ver, dtype: int64
--------------------
Android_Ver
False    9678
True        3
Name: Android_Ver, dtype: int64
--------------------
```

The NaN values will be dropped because the Rating column having the most NaN value is part of EDA

[16]:
```
df.dropna(inplace=True)

df.isna().sum()
```

[16]:
```
App               0
Category          0
Rating            0
Reviews           0
Size              0
Installs          0
Type              0
Price             0
Content_Rating    0
Genres            0
Last_Updated      0
Current_Ver       0
Android_Ver       0
dtype: int64
```

Now, I will clean the data column by column

Price column

[17]:
```
df.Price.unique()
```

[17]:
```
array(['0', '$4.99', '$3.99', '$6.99', '$7.99', '$5.99', '$2.99', '$3.49',
       '$1.99', '$9.99', '$7.49', '$0.99', '$9.00', '$5.49', '$10.00',
       '$24.99', '$11.99', '$79.99', '$16.99', '$14.99', '$29.99',
       '$12.99', '$2.49', '$10.99', '$1.50', '$19.99', '$15.99', '$33.99',
       '$39.99', '$3.95', '$4.49', '$1.70', '$8.99', '$1.49', '$3.88',
       '$399.99', '$17.99', '$400.00', '$3.02', '$1.76', '$4.84', '$4.77',
       '$1.61', '$2.50', '$1.59', '$6.49', '$1.29', '$299.99', '$379.99',
       '$37.99', '$18.99', '$389.99', '$8.49', '$1.75', '$14.00', '$2.00',
       '$3.08', '$2.59', '$19.40', '$3.90', '$4.59', '$15.46', '$3.04',
       '$13.99', '$4.29', '$3.28', '$4.60', '$1.00', '$2.95', '$2.90',
       '$1.97', '$2.56', '$1.20'], dtype=object)
```

I will strip the $ sign in the Price column t have the column in floating point

[18]:
```
df.Price = df.Price.apply(lambda x: x.replace('$', '').replace(',','')
                                    if isinstance(x, str) else x).astype(float)
```

[19]:
```
df.Price.unique()
```

[19]:
```
array([  0.  ,   4.99,   3.99,   6.99,   7.99,   5.99,   2.99,   3.49,
         1.99,   9.99,   7.49,   0.99,   9.  ,   5.49,  10.  ,  24.99,
         11.99,  79.99,  16.99,  14.99,  29.99,  12.99,   2.49,  10.99,
          1.5 ,  19.99,  15.99,  33.99,  39.99,   3.95,   4.49,   1.7 ,
          8.99,   1.49,   3.88, 399.99,  17.99, 400.  ,   3.02,   1.76,
          4.84,   4.77,   1.61,   2.5 ,   1.59,   6.49,   1.29, 299.99,
        379.99,  37.99,  18.99, 389.99,   8.49,   1.75,  14.  ,   2.  ,
          3.08,   2.59,  19.4 ,   3.9 ,   4.59,  15.46,   3.04,  13.99,
          4.29,   3.28,   4.6 ,   1.  ,   2.95,   2.9 ,   1.97,   2.56,
          1.2 ])
```

Let us check the changes

[20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8210 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   App             8210 non-null   object
 1   Category        8210 non-null   object
 2   Rating          8210 non-null   float64
 3   Reviews         8210 non-null   object
 4   Size            8210 non-null   object
 5   Installs        8210 non-null   object
 6   Type            8210 non-null   object
 7   Price           8210 non-null   float64
 8   Content_Rating  8210 non-null   object
 9   Genres          8210 non-null   object
 10  Last_Updated    8210 non-null   object
 11  Current_Ver     8210 non-null   object
 12  Android_Ver     8210 non-null   object
dtypes: float64(2), object(11)
memory usage: 898.0+ KB
```

[21]:
```
df.Reviews = df.Reviews.astype(int)

df.Reviews.unique()
```

[21]:
```
array([   159,    967,  87510, …,    603,   1195, 398307])
```

Size Column

[22]:
```
df.Size.unique()
```

[22]:
```
array(['19M', '14M', '8.7M', '25M', '2.8M', '5.6M', '29M', '33M', '3.1M',
       '28M', '12M', '20M', '21M', '37M', '5.5M', '17M', '39M', '31M',
       '4.2M', '23M', '6.0M', '6.1M', '4.6M', '9.2M', '5.2M', '11M',
       '24M', 'Varies with device', '9.4M', '15M', '10M', '1.2M', '26M',
       '8.0M', '7.9M', '56M', '57M', '35M', '54M', '201k', '3.6M', '5.7M',
       '8.6M', '2.4M', '27M', '2.7M', '2.5M', '7.0M', '16M', '3.4M',
       '8.9M', '3.9M', '2.9M', '38M', '32M', '5.4M', '18M', '1.1M',
       '2.2M', '4.5M', '9.8M', '52M', '9.0M', '6.7M', '30M', '2.6M',
       '7.1M', '22M', '6.4M', '3.2M', '8.2M', '4.9M', '9.5M', '5.0M',
       '5.9M', '13M', '73M', '6.8M', '3.5M', '4.0M', '2.3M', '2.1M',
       '42M', '9.1M', '55M', '23k', '7.3M', '6.5M', '1.5M', '7.5M', '51M',
       '41M', '48M', '8.5M', '46M', '8.3M', '4.3M', '4.7M', '3.3M', '40M',
       '7.8M', '8.8M', '6.6M', '5.1M', '61M', '66M', '79k', '8.4M',
       '3.7M', '118k', '44M', '695k', '1.6M', '6.2M', '53M', '1.4M',
       '3.0M', '7.2M', '5.8M', '3.8M', '9.6M', '45M', '63M', '49M', '77M',
       '4.4M', '70M', '9.3M', '8.1M', '36M', '6.9M', '7.4M', '84M', '97M',
       '2.0M', '1.9M', '1.8M', '5.3M', '47M', '556k', '526k', '76M',
       '7.6M', '59M', '9.7M', '78M', '72M', '43M', '7.7M', '6.3M', '334k',
       '93M', '65M', '79M', '100M', '58M', '50M', '68M', '64M', '34M',
       '67M', '60M', '94M', '9.9M', '232k', '99M', '624k', '95M', '8.5k',
       '41k', '292k', '80M', '1.7M', '10.0M', '74M', '62M', '69M', '75M',
       '98M', '85M', '82M', '96M', '87M', '71M', '86M', '91M', '81M',
       '92M', '83M', '88M', '704k', '862k', '899k', '378k', '4.8M',
       '266k', '375k', '1.3M', '975k', '980k', '4.1M', '89M', '696k',
       '544k', '525k', '920k', '779k', '853k', '720k', '713k', '772k',
       '318k', '58k', '241k', '196k', '857k', '51k', '953k', '865k',
       '251k', '930k', '540k', '313k', '746k', '203k', '26k', '314k',
       '239k', '371k', '220k', '730k', '756k', '91k', '293k', '17k',
       '74k', '14k', '317k', '78k', '924k', '818k', '81k', '939k', '169k',
       '45k', '965k', '90M', '545k', '61k', '283k', '655k', '714k', '93k',
       '872k', '121k', '322k', '976k', '206k', '954k', '444k', '717k',
       '210k', '609k', '308k', '306k', '175k', '350k', '383k', '454k',
       '1.0M', '70k', '812k', '442k', '842k', '417k', '412k', '459k',
       '478k', '335k', '782k', '721k', '430k', '429k', '192k', '460k',
       '728k', '496k', '816k', '414k', '506k', '887k', '613k', '778k',
       '683k', '592k', '186k', '840k', '647k', '373k', '437k', '598k',
       '716k', '585k', '982k', '219k', '55k', '323k', '691k', '511k',
       '951k', '963k', '25k', '554k', '351k', '27k', '82k', '208k',
       '551k', '29k', '103k', '116k', '153k', '209k', '499k', '173k',
       '597k', '809k', '122k', '411k', '400k', '801k', '787k', '50k',
       '643k', '986k', '516k', '837k', '780k', '20k', '498k', '600k',
       '656k', '221k', '228k', '176k', '34k', '259k', '164k', '458k',
       '629k', '28k', '288k', '775k', '785k', '636k', '916k', '994k',
       '309k', '485k', '914k', '903k', '608k', '500k', '54k', '562k',
       '847k', '948k', '811k', '270k', '48k', '523k', '784k', '280k',
       '24k', '892k', '154k', '18k', '33k', '860k', '364k', '387k',
       '626k', '161k', '879k', '39k', '170k', '141k', '160k', '144k',
       '143k', '190k', '376k', '193k', '473k', '246k', '73k', '253k',
       '957k', '420k', '72k', '404k', '470k', '226k', '240k', '89k',
       '234k', '257k', '861k', '467k', '676k', '552k', '582k', '619k'],
      dtype=object)
```

For Size column to make better sense, I will strip the M and k and covert the values to Kilobyte

[23]:
```
df = df[df.Size != 'Varies with device']
```

[24]:
```
df['Si'] = df.Size.str[:-1].astype(float)
df['KM'] = df.Size.str[-1]
df['KM'] = df.KM.replace(['k', 'M'], [1, 1024])
```

[25]:
```
df.KM.unique()
```

[25]:
```
array([1024,    1])
```

[26]:
```
df.Size = df.KM * df.Si
```

[27]:
```
df.Installs.unique()
```

[27]:
```
array(['10,000+', '500,000+', '5,000,000+', '50,000,000+', '100,000+',
       '50,000+', '1,000,000+', '10,000,000+', '5,000+', '100,000,000+',
       '1,000+', '500,000,000+', '100+', '500+', '10+', '1,000,000,000+',
       '5+', '50+', '1+'], dtype=object)
```

[28]:
```
df.Type.unique()
```

[28]:
```
array(['Free', 'Paid'], dtype=object)
```

[29]:
```
df[df.Rating > 5]
```

[29]:
```
Empty DataFrame
Columns: [App, Category, Rating, Reviews, Size, Installs, Type, Price,
Content_Rating, Genres, Last_Updated, Current_Ver, Android_Ver, Si, KM]
Index: []
```

Recode the install column

[30]:
```
df.Installs.unique()
```

[30]:
```
array(['10,000+', '500,000+', '5,000,000+', '50,000,000+', '100,000+',
       '50,000+', '1,000,000+', '10,000,000+', '5,000+', '100,000,000+',
       '1,000+', '500,000,000+', '100+', '500+', '10+', '1,000,000,000+',
       '5+', '50+', '1+'], dtype=object)
```

[31]:
```
df.Installs = df.Installs.map({'10,000+' : 10000, '1,000+' : 1000, '5,000+' :␣
 ↪5000, '50,000+' : 50000, '100,000+': 100000, '100+': 100,
                          '10+' : 10, '500+' : 500, '50+' :50, '10,000,000+':
 ↪100000000, '1,000,000+':1000000, '500,000+' : 500000,
                          '1+' : 1, '5,000,000+' : 5000000, '50,000,000+' :␣
 ↪50000000, '100,000,000+': 100000000,
                          '1,000,000,000+' : 1000000000, '500,000,000+' :␣
↪5000000000, '5+' : 5})
```

[32]:
```
df.drop(columns= ['Si', 'KM'], inplace=True)
```

Now, Let us check our data very well before moving to EDA

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

[33]:
```
                                                 App        Category  Rating  \
0     Photo Editor & Candy Camera & Grid & ScrapBook  ART_AND_DESIGN     4.1
1                                Coloring book moana  ART_AND_DESIGN     3.9
2  U Launcher Lite – FREE Live Cool Themes, Hide …  ART_AND_DESIGN     4.7
3                              Sketch - Draw & Paint  ART_AND_DESIGN     4.5
4              Pixel Draw - Number Art Coloring Book  ART_AND_DESIGN     4.3

  Reviews     Size  Installs  Type  Price Content_Rating  \
0     159  19456.0     10000  Free    0.0       Everyone
1     967  14336.0    500000  Free    0.0       Everyone
2   87510   8908.8   5000000  Free    0.0       Everyone
3  215644  25600.0  50000000  Free    0.0           Teen
4     967   2867.2    100000  Free    0.0       Everyone

                      Genres      Last_Updated         Current_Ver  \
0               Art & Design   January 7, 2018               1.0.0
1  Art & Design;Pretend Play  January 15, 2018               2.0.0
2               Art & Design    August 1, 2018               1.2.4
3               Art & Design      June 8, 2018  Varies with device
4    Art & Design;Creativity     June 20, 2018                 1.1

   Android_Ver
0  4.0.3 and up
1  4.0.3 and up
2  4.0.3 and up
3    4.2 and up
4    4.4 and up
```

[34]:
```
df.describe().transpose()
```

[34]:
```
           count          mean           std  min      25%       50%  \
Rating    7039.0  4.161202e+00  5.587983e-01  1.0      4.0       4.3
Reviews   7039.0  1.459494e+05  1.027101e+06  1.0     85.0    1563.0
Size      7039.0  2.228158e+04  2.326649e+04  8.5   5017.6   13312.0
Installs  7039.0  1.791602e+07  1.717069e+08  1.0  10000.0  100000.0
Price     7039.0  1.171219e+00  1.818216e+01  0.0      0.0       0.0

                75%          max
```

```
Rating          4.5  5.000000e+00
Reviews     26848.0  4.489172e+07
Size        31744.0  1.024000e+05
Installs  1000000.0  5.000000e+09
Price           0.0  4.000000e+02
```

[35]:
```
jovian.commit(project='GooglePlayStore_Project', environment=None)
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "huygens123/googleplaystore-project" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/huygens123/googleplaystore-
project
```

[35]:
```
'https://jovian.ai/huygens123/googleplaystore-project'
```

4 Exploratory Data Analysis

- Let us check the type of data set available

[36]:
```
#for categorical variable
categorical = [variable for variable in df.columns
            if df[variable].dtype == 'O']

# for numerical variable
numerical = [variable for variable in df.columns
           if df[variable].dtype != 'O']

print('The number of categorical variables in the dataset is: {}'.
 ↪format(len(categorical)))
print('The number of numerical variables in the dataset is: {}'.
 ↪format(len(numerical)))
```

```
The number of categorical variables in the dataset is: 8
The number of numerical variables in the dataset is: 5
```

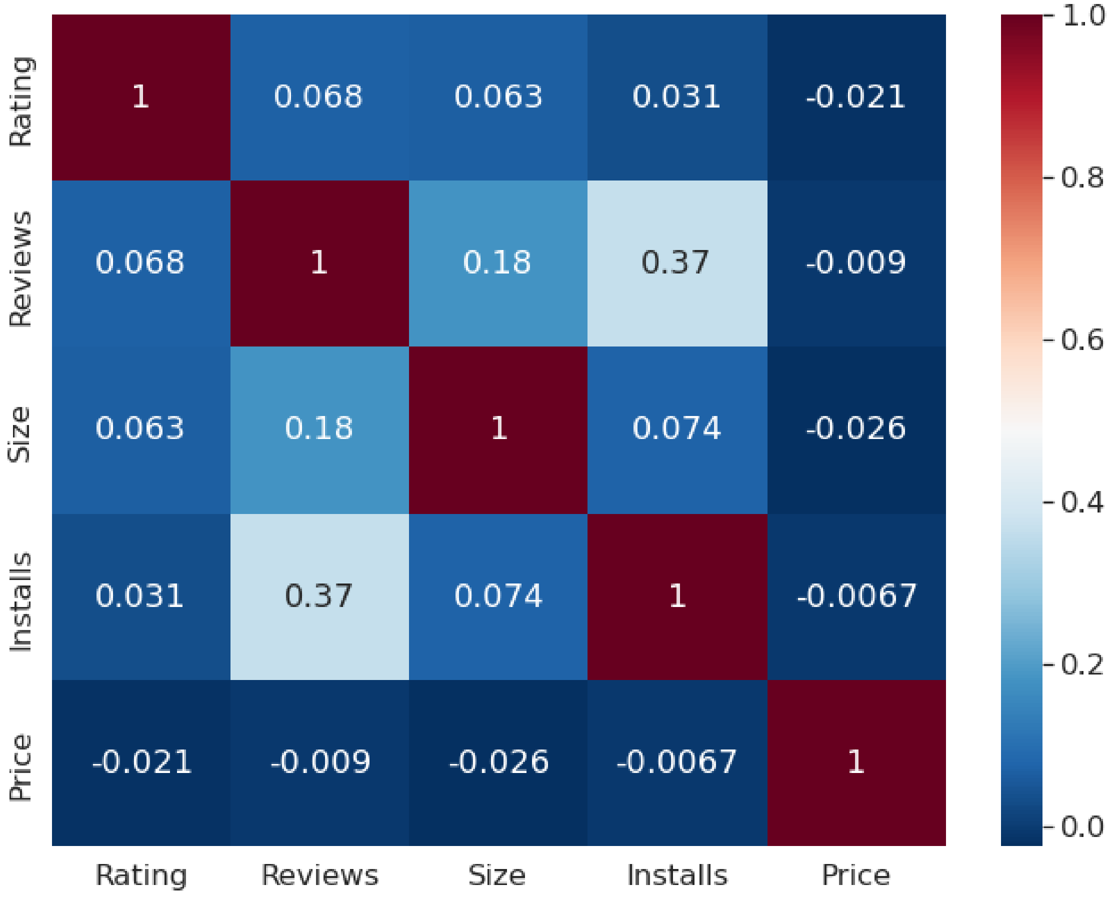
What is the correlation among the numerical variables?

[37]:
```
df.corr()
```

[37]:
```
            Rating   Reviews      Size  Installs     Price
Rating    1.000000  0.067801  0.062779  0.031256 -0.021205
Reviews   0.067801  1.000000  0.181866  0.365966 -0.009030
Size      0.062779  0.181866  1.000000  0.074399 -0.025697
Installs  0.031256  0.365966  0.074399  1.000000 -0.006709
Price    -0.021205 -0.009030 -0.025697 -0.006709  1.000000
```

Let us vizualise this using heatmap

[38]:
```
plt.figure(figsize=(10, 8))
sns.heatmap(df.corr(), cmap = 'RdBu_r', annot= True);
```



Installs column and Reviews column have the best correlation followed by Size and Review

Let us check and vizualise how the type of App are distributed i.e free or paid

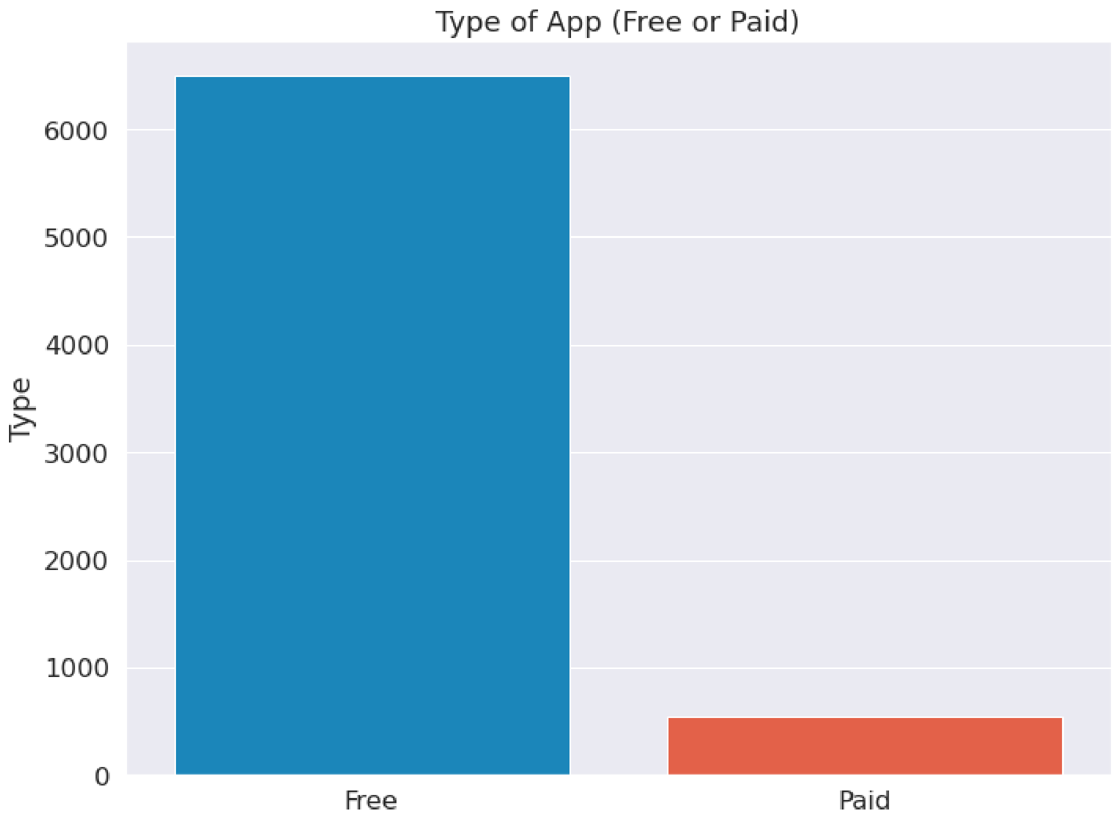
[39]:
```
Type = df.Type.value_counts()
Type
```

[39]:
```
Free    6500
Paid     539
Name: Type, dtype: int64
```

So, there are 6,500 apps that are free while the other 539 are paid app

[40]:

```
plt.figure(figsize=(10, 8))
sns.barplot(x = Type.index, y = Type);
plt.title("Type of App (Free or Paid)");
```



4.0.1 Category

First, let's check the number of App in each Category

[41]:

```
Category = df.Category.value_counts()
Category
```

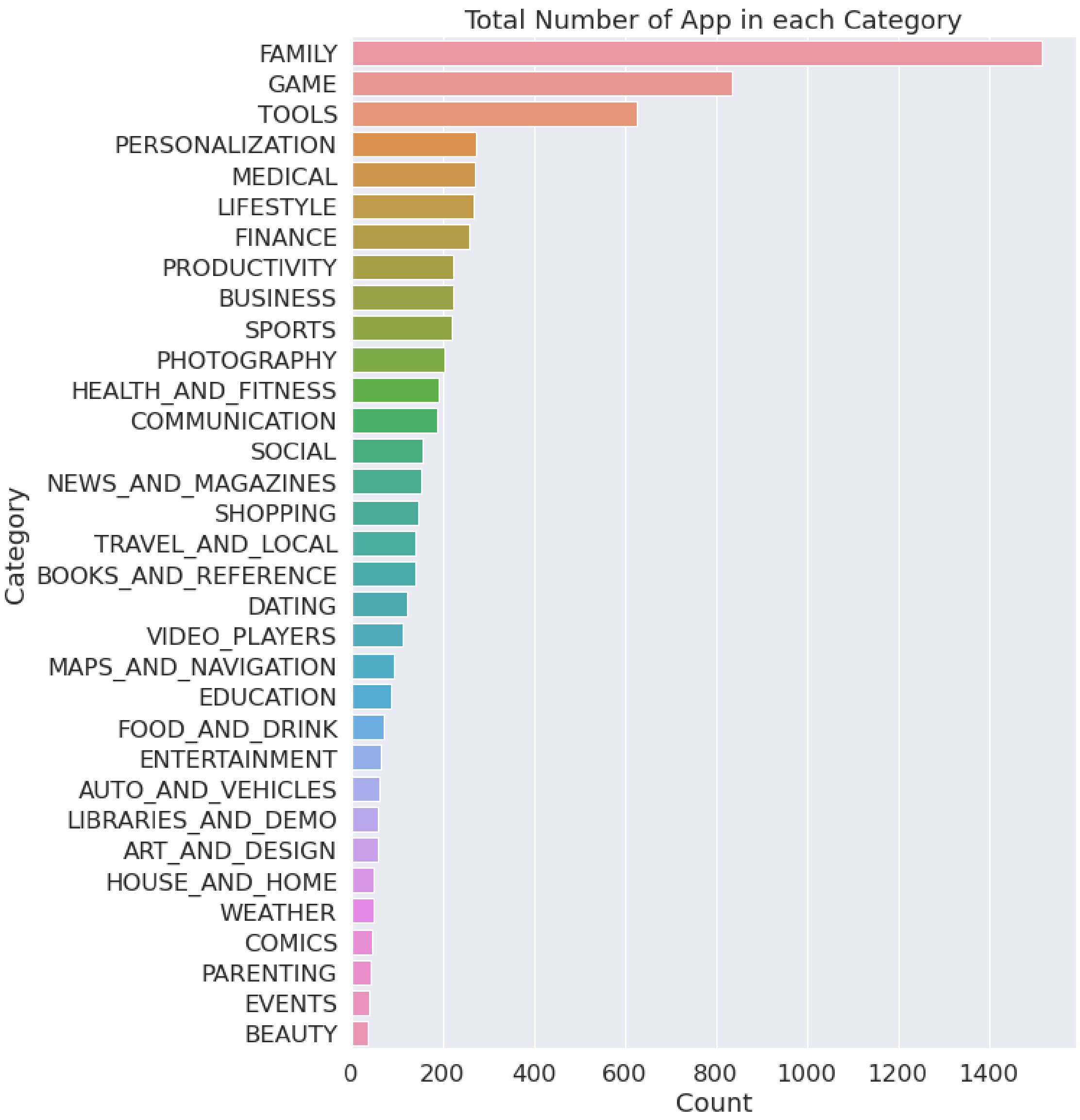
[41]:

```
FAMILY               1515
GAME                  835
TOOLS                 627
PERSONALIZATION       274
MEDICAL               270
LIFESTYLE             269
FINANCE               258
PRODUCTIVITY          224
BUSINESS              222
SPORTS                221
```

```
PHOTOGRAPHY             206
HEALTH_AND_FITNESS      191
COMMUNICATION           189
SOCIAL                  156
NEWS_AND_MAGAZINES      154
SHOPPING                146
TRAVEL_AND_LOCAL        141
BOOKS_AND_REFERENCE     141
DATING                  122
VIDEO_PLAYERS           113
MAPS_AND_NAVIGATION      94
EDUCATION                88
FOOD_AND_DRINK           72
ENTERTAINMENT            64
AUTO_AND_VEHICLES        63
LIBRARIES_AND_DEMO       60
ART_AND_DESIGN           58
HOUSE_AND_HOME           50
WEATHER                  50
COMICS                   47
PARENTING                44
EVENTS                   38
BEAUTY                   37
Name: Category, dtype: int64
```

Family has the highest number of App (1515) while Beauty is lowest represented (37)

[42]:
```
plt.figure(figsize=(8, 12))
sns.barplot(y = Category.index, x = Category)
plt.ylabel('Category')
plt.xlabel('Count')
plt.title("Total Number of App in each Category");
```



4.0.2 Content Ratings

Next, Let us check the distribution of `Content_Rating` column

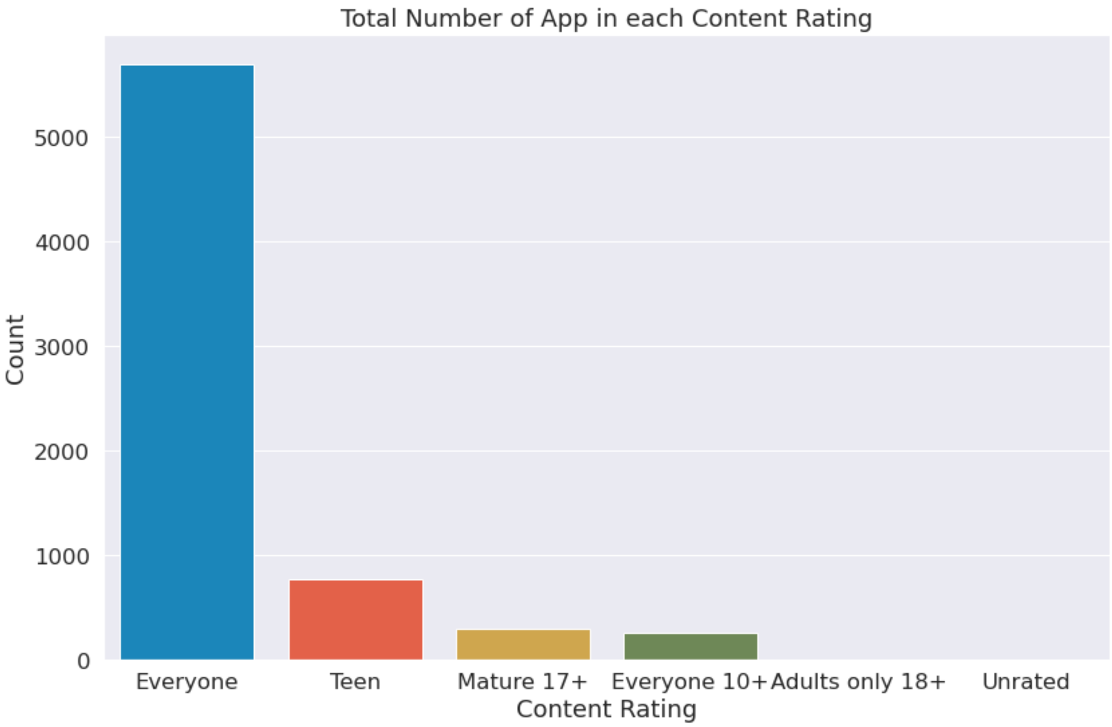
[43]:
```
Content_Rating = df.Content_Rating.value_counts()
Content_Rating
```

[43]:
```
Everyone           5694
Teen                779
Mature 17+          304
Everyone 10+        259
Adults only 18+       2
```

```
Unrated                  1
Name: Content_Rating, dtype: int64
```

Majority of the Apps is either for Everyone or for Teen. Let us vizualise this with bar chart

[44]:
```
plt.figure(figsize= (12, 8))
sns.barplot(x = Content_Rating.index, y = Content_Rating)
plt.ylabel('Count')
plt.xlabel('Content Rating')
plt.title("Total Number of App in each Content Rating");
```



4.0.3 Genres

[45]:
```
Genre  = df.Genres.value_counts()
Genre
```

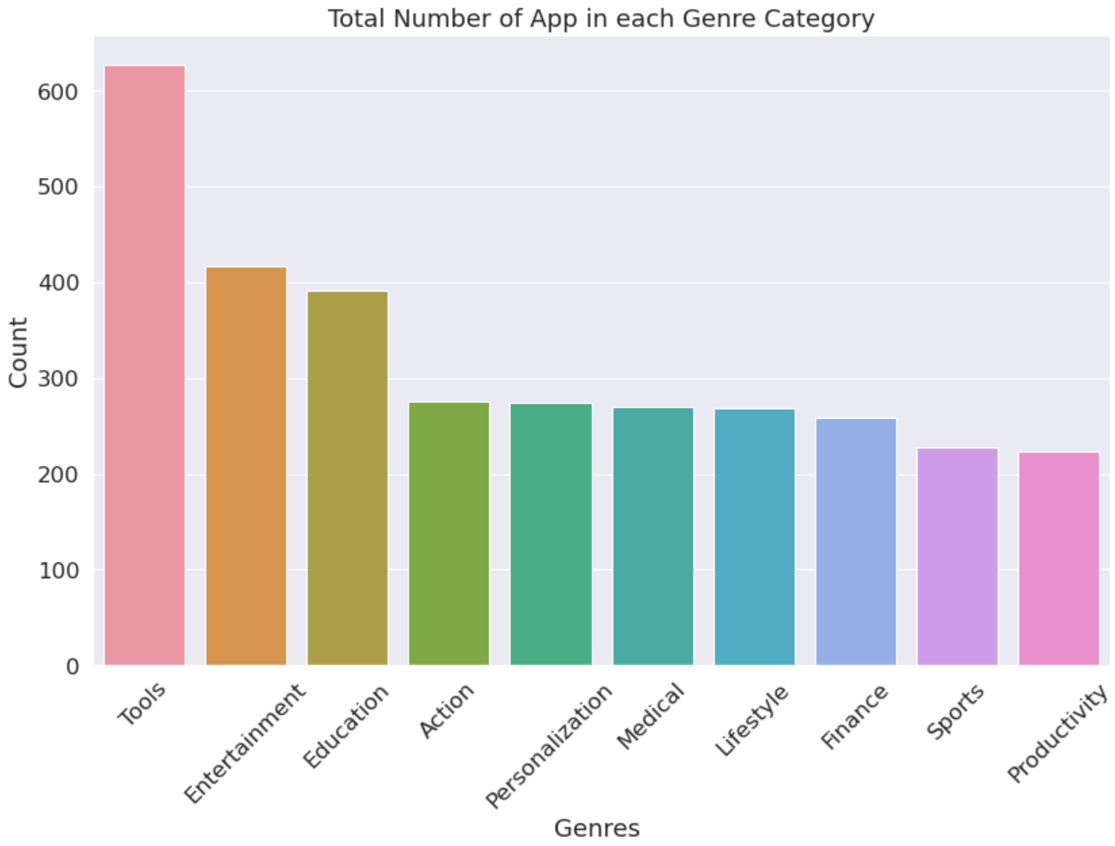
[45]:
```
Tools                     627
Entertainment             417
Education                 392
Action                    276
Personalization           274
                         …
Card;Brain Games            1
```

```
Lifestyle;Pretend Play      1
Education;Brain Games       1
Comics;Creativity           1
Strategy;Creativity         1
Name: Genres, Length: 112, dtype: int64
```

Tools app and Entertainment app take the lead while Strategy and Creativity app are least represented

[46]:
```python
# vizualie to 10 Genres
Genre = Genre.sort_values(ascending=False).head(10)
```

[47]:
```python
plt.figure(figsize= (12, 8))
sns.barplot(x = Genre.index, y = Genre)
plt.ylabel('Count')
plt.xlabel('Genres ')
plt.title("Total Number of App in each Genre Category")
plt.xticks(rotation = 45);
```

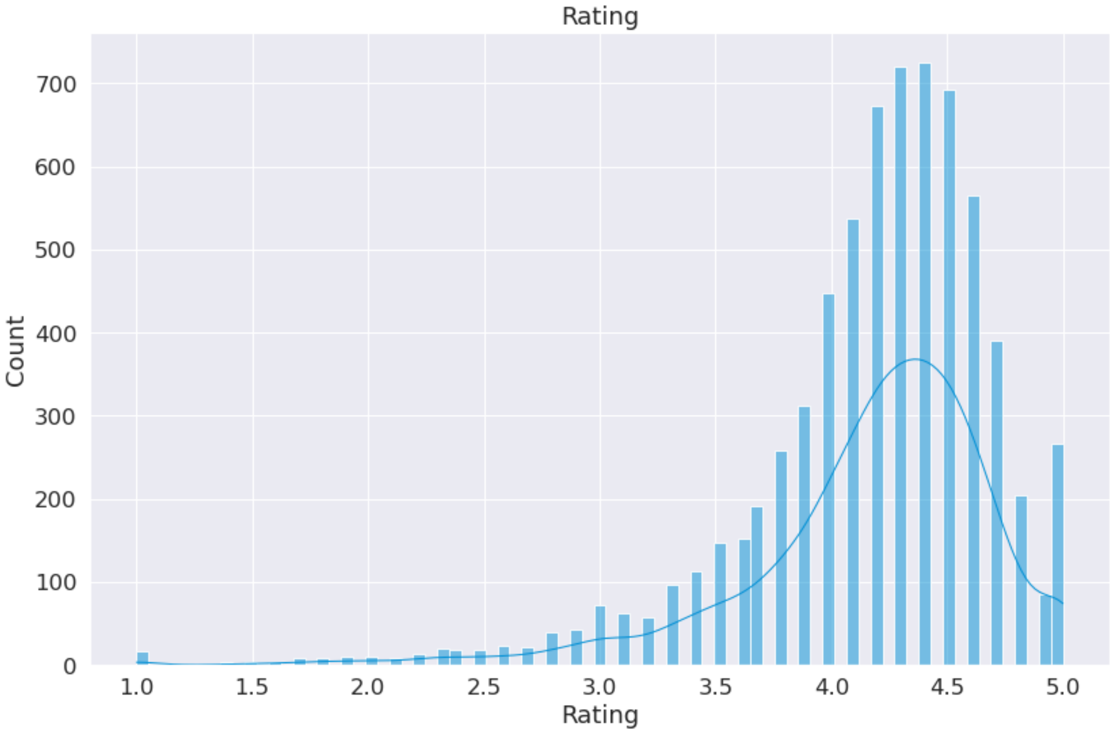


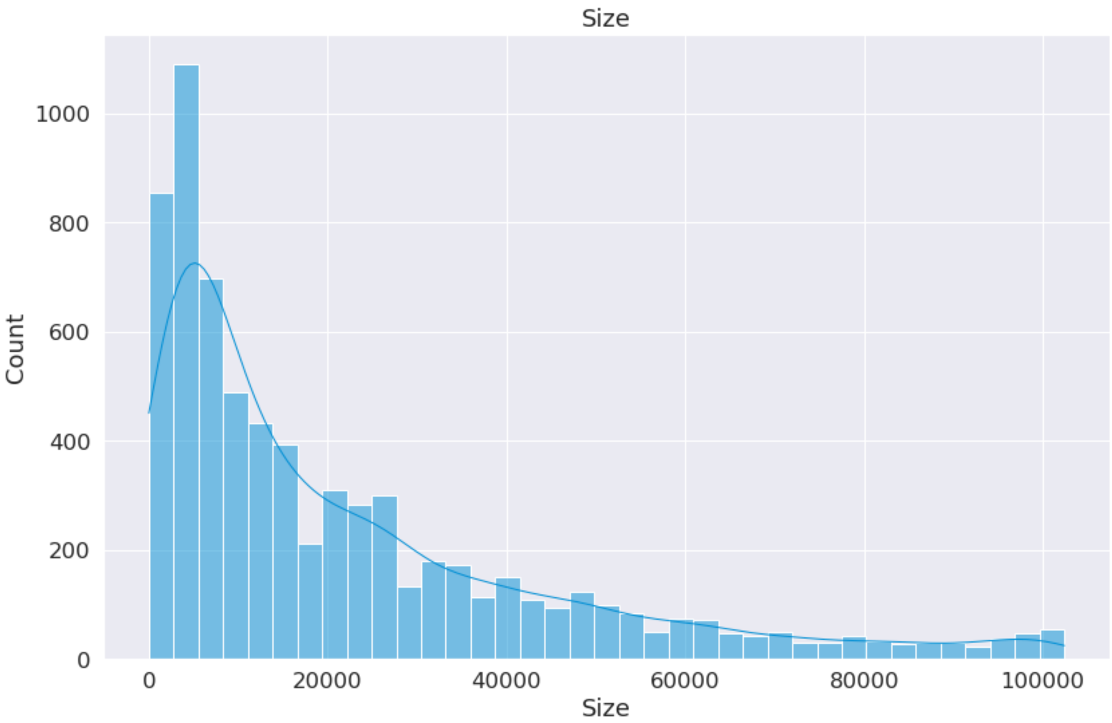
Next, I will show the distribution of continuous values

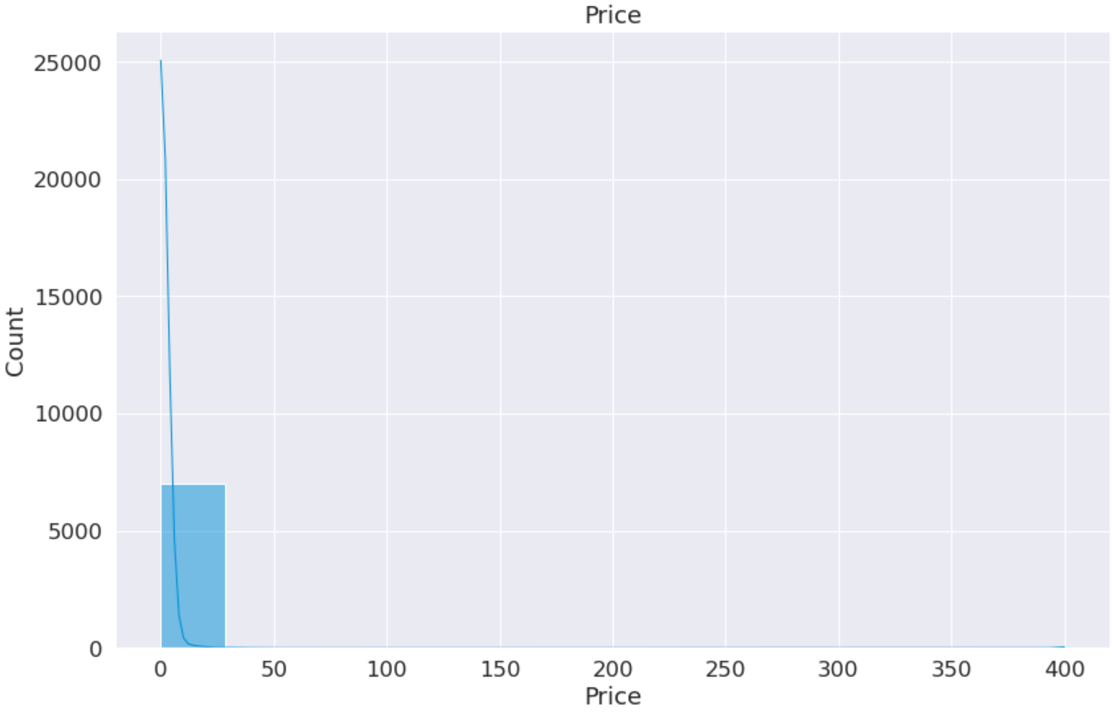
[48]:

```
# Plotting continous features
continous_features = df.select_dtypes(['float64']).columns.to_list()

for cont_feature in continous_features:
    plt.figure(figsize= (12, 8))
    ax = sns.histplot(df[cont_feature], kde = True)
    plt.title(cont_feature);
```







Ratings column is skewed to the right while the `Size` column is skewed to the right.

The impplication of this is that, there are higher ratings in the dataset than low rating. Consequently, the `Size` column shows that there are more smaller (size) app in the dataset than heavier app

5 QUESTIONS

In this section, I will ask some questions based on EDA above and answer them

List of top 10 most installed and Rating `App`

```
[49]: App = df.groupby('App')[['Installs', 'Rating', 'Size', 'Price', 'Reviews']].
       ↪mean()
      App
```

```
[49]:                                                 Installs  Rating     Size  \
      App
      +Download 4 Instagram Twitter                  1000000.0     4.5  22528.0
      - Free Comics - Comic Apps                       10000.0     3.5   9318.4
      .R                                               10000.0     4.5    203.0
      /u/app                                           10000.0     4.7  54272.0
      058.ba                                             100.0     4.4  14336.0
      …                                                      …       …        …
         BL                                         10000.0     4.4  34816.0
        I'm rich                                       10000.0     3.8  26624.0
        WhatsLov: Smileys of love, stickers and GIF  1000000.0     4.6  18432.0
        Smart Ruler   cm/inch measuring for homework!  10000.0     4.0   3276.8
        Football Wallpapers 4K | Full HD Backgrounds  1000000.0     4.7   4096.0

                                                      Price  Reviews
      App
      +Download 4 Instagram Twitter                    0.00  40467.0
      - Free Comics - Comic Apps                       0.00    115.0
      .R                                               0.00    259.0
      /u/app                                           0.00    573.0
      058.ba                                           0.00     27.0
      …                                                   …        …
         BL                                         0.00    190.0
        I'm rich                                       399.99    718.0
        WhatsLov: Smileys of love, stickers and GIF     0.00  22098.0
        Smart Ruler   cm/inch measuring for homework!  0.00     19.0
        Football Wallpapers 4K | Full HD Backgrounds    0.00  11661.0

      [7023 rows x 5 columns]
```

Most Installed App

```
[50]: App_installs = App.sort_values('Installs', ascending=False).head(10)
      App_installs
```

[50]:

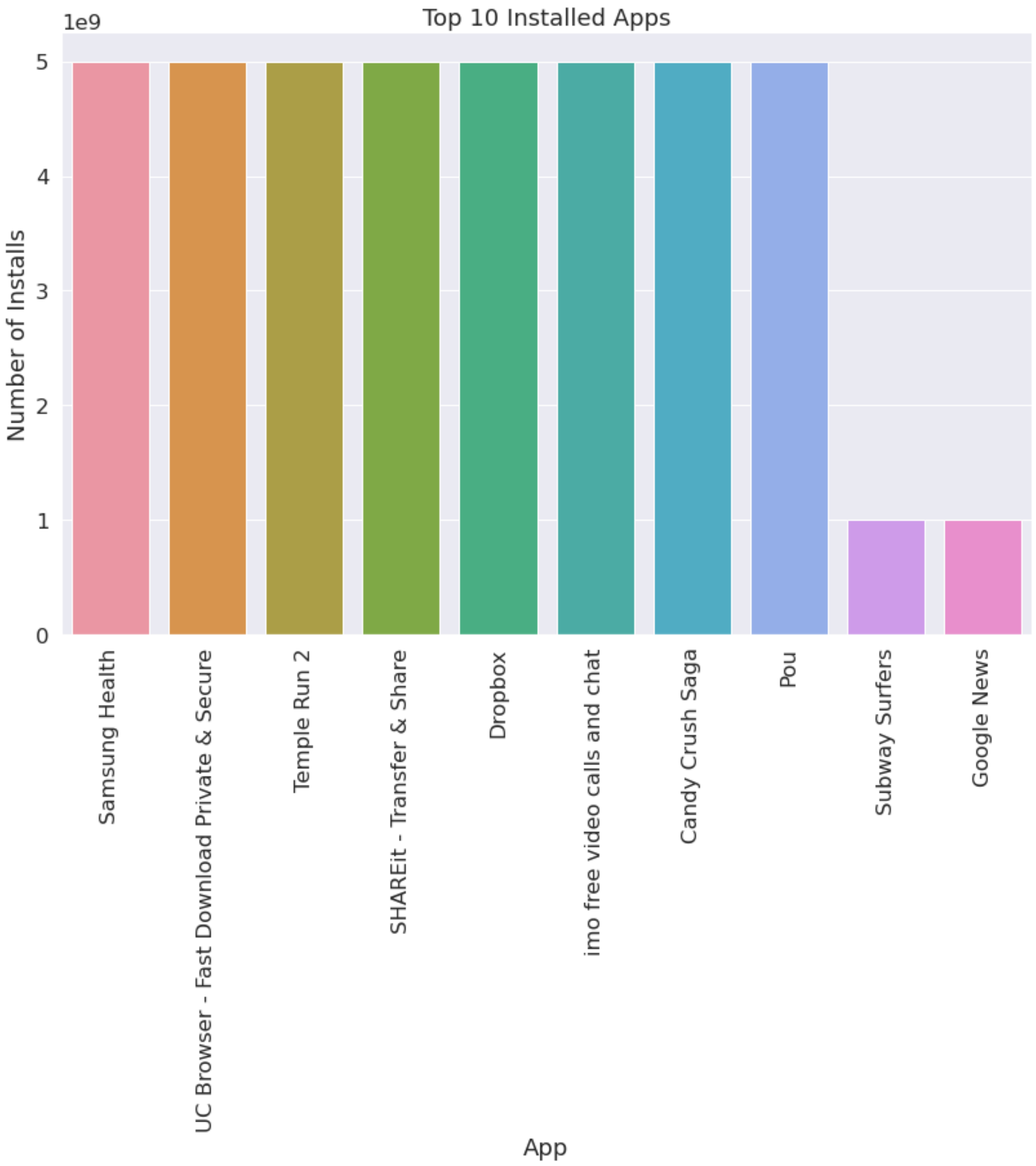
```
                                                 Installs  Rating     Size  \
App
Samsung Health                               5.000000e+09     4.3  71680.0
UC Browser - Fast Download Private & Secure  5.000000e+09     4.5  40960.0
Temple Run 2                                 5.000000e+09     4.3  63488.0
SHAREit - Transfer & Share                   5.000000e+09     4.6  17408.0
Dropbox                                      5.000000e+09     4.4  62464.0
imo free video calls and chat                5.000000e+09     4.3  11264.0
Candy Crush Saga                             5.000000e+09     4.4  75776.0
Pou                                          5.000000e+09     4.3  24576.0
Subway Surfers                               1.000000e+09     4.5  77824.0
Google News                                  1.000000e+09     3.9  13312.0

                                             Price     Reviews
App
Samsung Health                                 0.0    480208.0
UC Browser - Fast Download Private & Secure    0.0  17712922.0
Temple Run 2                                   0.0   8118609.0
SHAREit - Transfer & Share                     0.0   7790693.0
Dropbox                                        0.0   1861310.0
imo free video calls and chat                  0.0   4785892.0
Candy Crush Saga                               0.0  22426677.0
Pou                                            0.0  10485308.0
Subway Surfers                                 0.0  27722264.0
Google News                                    0.0    877635.0
```

Clearly, Samsung Health app has the most total installs

[51]:

```
plt.figure(figsize= (12, 8))
sns.barplot(x = App_installs.index, y = App_installs.Installs)
plt.title("Top 10 Installed Apps")
plt.ylabel("Number of Installs")
plt.xticks(rotation = 90);
```



Most Reviewed App

[52]:

```
App_Review = App.sort_values('Reviews', ascending=False).head(10)
App_Review
```

[52]:

```
                                                        Installs  Rating  \
App
Clash of Clans                                      1.000000e+08     4.6
Subway Surfers                                      1.000000e+09     4.5
```

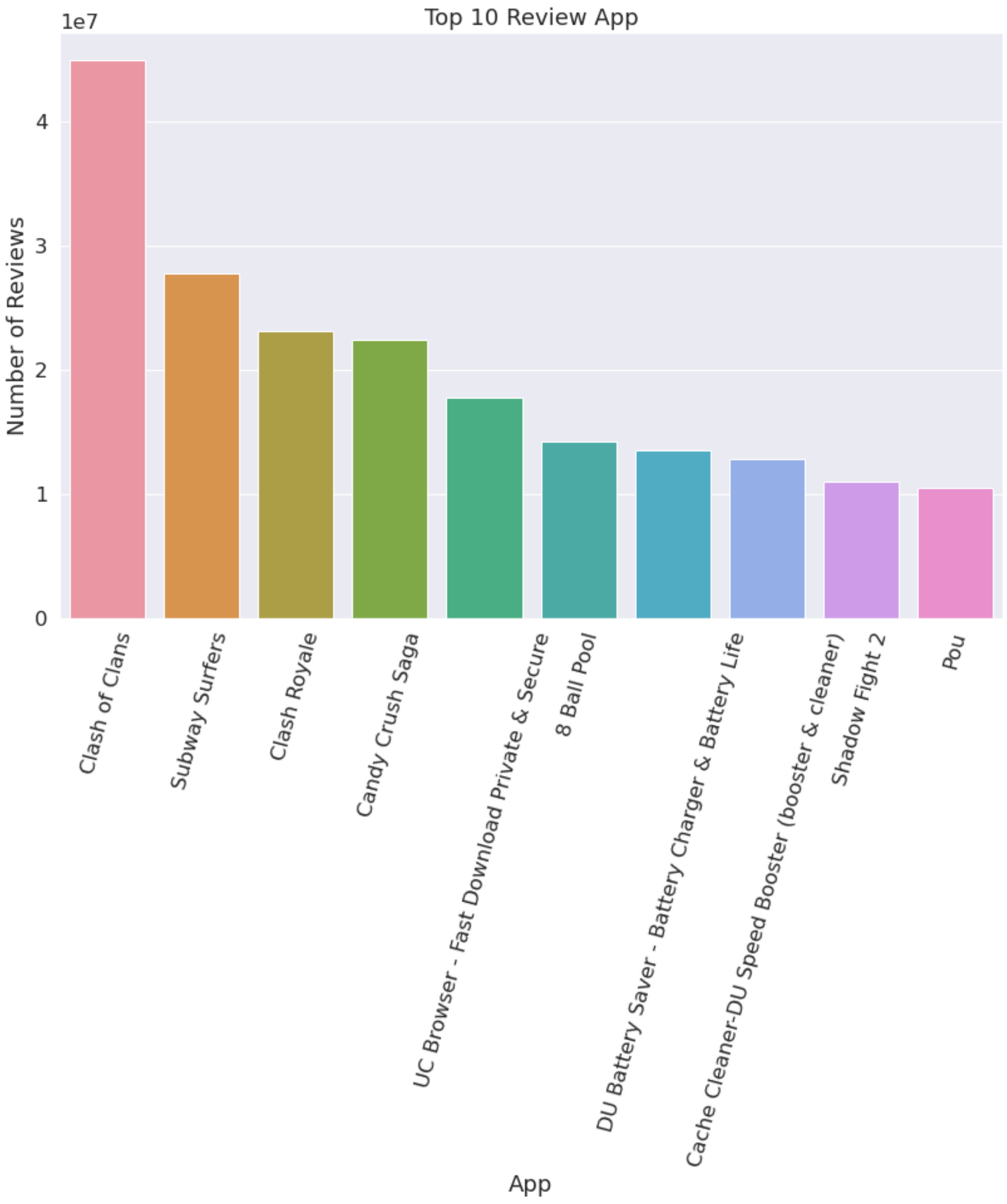
```
Clash Royale                                    1.000000e+08     4.6
Candy Crush Saga                                5.000000e+09     4.4
UC Browser - Fast Download Private & Secure     5.000000e+09     4.5
8 Ball Pool                                     1.000000e+08     4.5
DU Battery Saver - Battery Charger & Battery Life  1.000000e+08     4.5
Cache Cleaner-DU Speed Booster (booster & cleaner) 1.000000e+08     4.5
Shadow Fight 2                                  1.000000e+08     4.6
Pou                                             5.000000e+09     4.3

                                                       Size  Price  \
App
Clash of Clans                                     100352.0    0.0
Subway Surfers                                      77824.0    0.0
Clash Royale                                        99328.0    0.0
Candy Crush Saga                                    75776.0    0.0
UC Browser - Fast Download Private & Secure         40960.0    0.0
8 Ball Pool                                         53248.0    0.0
DU Battery Saver - Battery Charger & Battery Life   14336.0    0.0
Cache Cleaner-DU Speed Booster (booster & cleaner)  15360.0    0.0
Shadow Fight 2                                      90112.0    0.0
Pou                                                 24576.0    0.0

                                                      Reviews
App
Clash of Clans                                     44891723.0
Subway Surfers                                     27722264.0
Clash Royale                                       23133508.0
Candy Crush Saga                                   22426677.0
UC Browser - Fast Download Private & Secure        17712922.0
8 Ball Pool                                        14198297.0
DU Battery Saver - Battery Charger & Battery Life  13479633.0
Cache Cleaner-DU Speed Booster (booster & cleaner) 12759663.0
Shadow Fight 2                                     10979062.0
Pou                                                10485308.0
```

Clash of Clans has the most reviews of 100352

[53]:
```
plt.figure(figsize= (12, 8))
sns.barplot(x = App_Review.index, y = App_Review.Reviews)
plt.title("Top 10 Review App")
plt.ylabel("Number of Reviews")
plt.xticks(rotation = 75);
```



Most Cost App

[54]:
```
App_Price = App.sort_values('Price', ascending=False).head(10)
App_Price
```

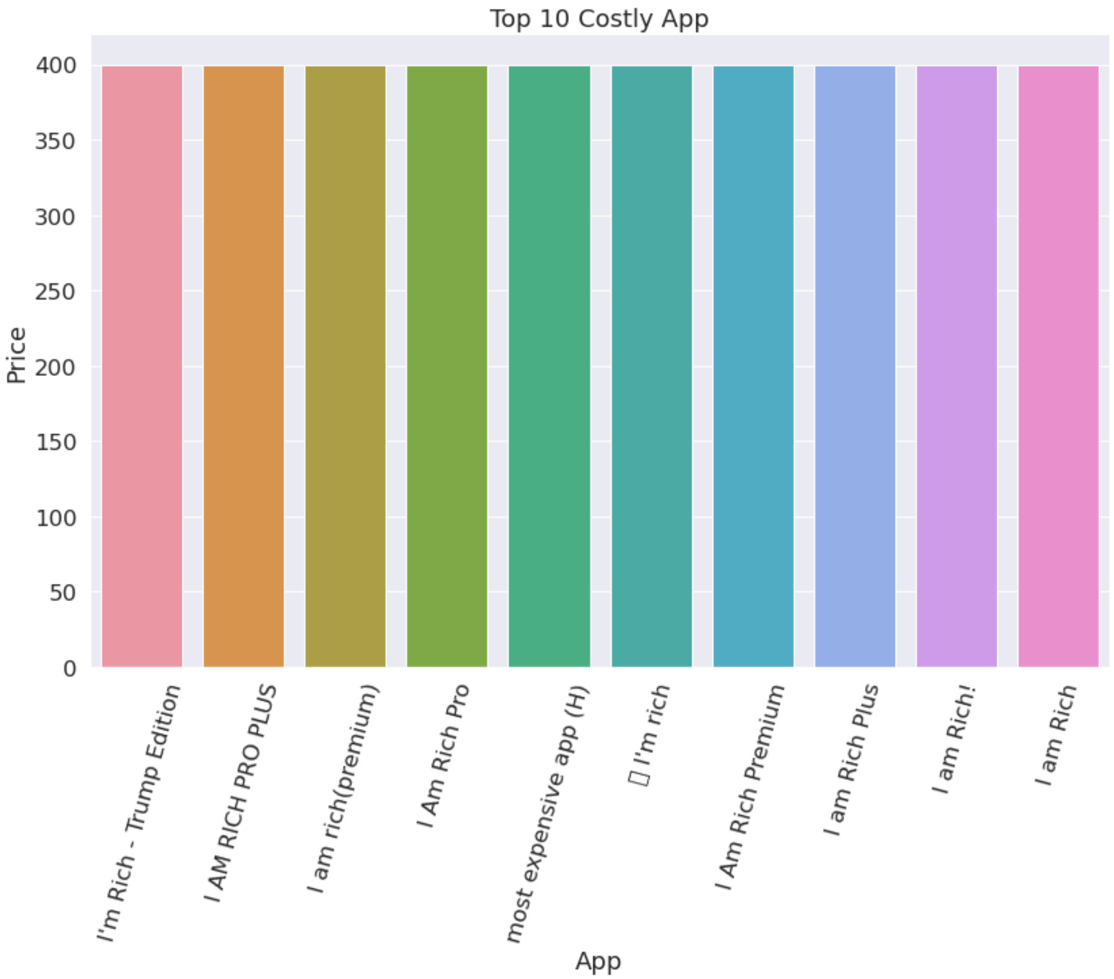
[54]:
```
                                 Installs  Rating     Size   Price  Reviews
App
```

```
I'm Rich - Trump Edition   10000.0      3.6   7475.2  400.00     275.0
I AM RICH PRO PLUS          1000.0      4.0  41984.0  399.99      36.0
I am rich(premium)          5000.0      3.5    965.0  399.99     472.0
I Am Rich Pro               5000.0      4.4   2764.8  399.99     201.0
most expensive app (H)       100.0      4.3   1536.0  399.99       6.0
  I'm rich                10000.0      3.8  26624.0  399.99     718.0
I Am Rich Premium          50000.0      4.1   4812.8  399.99    1867.0
I am Rich Plus             10000.0      4.0   8908.8  399.99     856.0
I am Rich!                  1000.0      3.8  22528.0  399.99      93.0
I am Rich                   5000.0      4.3   3891.2  399.99     180.0
```

I'm Rich - Trump Edition is the most costly app priced at 400 Dollars

[55]:
```python
plt.figure(figsize= (12, 8))
sns.barplot(x = App_Price.index, y = App_Price.Price)
plt.title("Top 10 Costly App")
plt.ylabel("Price")
plt.xticks(rotation = 75);
```

```
/opt/conda/lib/python3.9/site-packages/matplotlib/backends/backend_agg.py:240:
RuntimeWarning: Glyph 128142 missing from current font.
  font.set_text(s, 0.0, flags=flags)
/opt/conda/lib/python3.9/site-packages/matplotlib/backends/backend_agg.py:203:
RuntimeWarning: Glyph 128142 missing from current font.
  font.set_text(s, 0, flags=flags)
```



5.0.1 Top 10 Categories

```
[56]: Category_rate = df.groupby('Category')[['Installs', 'Rating', 'Size',␣
        ↪'Reviews', 'Price']].mean()
      Category_rate
```

```
[56]:                          Installs    Rating          Size         Reviews  \
      Category
      ART_AND_DESIGN        6.366002e+06  4.381034  13250.206897    18745.172414
      AUTO_AND_VEHICLES     3.551902e+06  4.147619  22058.200000    15750.571429
      BEAUTY                3.626000e+05  4.291892  15885.837838     5020.243243
      BOOKS_AND_REFERENCE   4.643859e+06  4.322695  14506.587234    23138.234043
      BUSINESS              4.577851e+06  4.096396  14268.742342    19777.873874
      COMICS                3.708745e+05  4.168085  13789.757447    13048.404255
      COMMUNICATION         6.867151e+07  4.076720  12226.704762   242959.502646
      DATING                2.906495e+06  3.963934  18082.832787    18008.622951
```

```
EDUCATION           8.945477e+06  4.373864  19456.659091   47422.409091
ENTERTAINMENT       2.824938e+07  4.154687  23596.800000  157911.734375
EVENTS              2.802211e+05  4.478947  14778.642105    3321.605263
FAMILY              9.071130e+06  4.180330  29682.500330   79913.601320
FINANCE             4.087548e+06  4.104651  18395.052713   34377.201550
FOOD_AND_DRINK      1.314539e+07  4.109722  22964.622222   46177.527778
GAME                4.773336e+07  4.236886  43845.124311  628736.355689
HEALTH_AND_FITNESS  3.364637e+07  4.191099  22459.813613   36209.261780
HOUSE_AND_HOME      4.629640e+06  4.128000  18102.272000   19833.040000
LIBRARIES_AND_DEMO  5.333050e+06  4.205000  12364.378333   14424.200000
LIFESTYLE           5.853231e+06  4.089963  15880.507063   28508.107807
MAPS_AND_NAVIGATION 8.553357e+06  4.008511  17589.055319   38399.521277
MEDICAL             1.246673e+05  4.167407  19254.183704    4303.003704
NEWS_AND_MAGAZINES  1.533052e+07  4.143506  12713.563636   38670.259740
PARENTING           2.581046e+06  4.347727  22097.454545   19999.500000
PERSONALIZATION     1.147858e+07  4.324453  11856.904380  102438.248175
PHOTOGRAPHY         2.526162e+07  4.114078  16239.429126  188123.708738
PRODUCTIVITY        3.272903e+07  4.131250  12313.932143  103773.040179
SHOPPING            2.336117e+07  4.213014  16932.819178  118748.712329
SOCIAL              1.280923e+07  4.257692  17728.923077  131664.653846
SPORTS              1.342222e+07  4.200905  27033.515837  113725.389140
TOOLS               1.925128e+07  4.007177   8484.219458  133337.633174
TRAVEL_AND_LOCAL    9.919421e+06  4.011348  25367.004255   31647.326241
VIDEO_PLAYERS       2.121028e+07  4.019469  16281.253097  204681.946903
WEATHER             9.785930e+06  4.242000  13223.288000   74133.900000

                       Price
Category
ART_AND_DESIGN      0.102931
AUTO_AND_VEHICLES   0.000000
BEAUTY              0.000000
BOOKS_AND_REFERENCE 0.148156
BUSINESS            0.231622
COMICS              0.000000
COMMUNICATION       0.220794
DATING              0.122787
EDUCATION           0.204091
ENTERTAINMENT       0.046719
EVENTS              0.000000
FAMILY              1.464310
FINANCE             9.456860
FOOD_AND_DRINK      0.069306
GAME                0.325018
HEALTH_AND_FITNESS  0.180209
HOUSE_AND_HOME      0.000000
LIBRARIES_AND_DEMO  0.000000
LIFESTYLE           7.261710
```

```
MAPS_AND_NAVIGATION  0.159149
MEDICAL              2.166519
NEWS_AND_MAGAZINES   0.025844
PARENTING            0.113409
PERSONALIZATION      0.420876
PHOTOGRAPHY          0.322184
PRODUCTIVITY         0.236429
SHOPPING             0.037534
SOCIAL               0.012692
SPORTS               0.363032
TOOLS                0.292823
TRAVEL_AND_LOCAL     0.188014
VIDEO_PLAYERS        0.008761
WEATHER              0.468800
```

Installations by Category

[57]:
```
Category_rate_Install = Category_rate.sort_values('Installs', ascending=False)
Category_rate_Install
```

[57]:
```
                         Installs    Rating          Size        Reviews  \
Category
COMMUNICATION        6.867151e+07  4.076720  12226.704762  242959.502646
GAME                 4.773336e+07  4.236886  43845.124311  628736.355689
HEALTH_AND_FITNESS   3.364637e+07  4.191099  22459.813613   36209.261780
PRODUCTIVITY         3.272903e+07  4.131250  12313.932143  103773.040179
ENTERTAINMENT        2.824938e+07  4.154687  23596.800000  157911.734375
PHOTOGRAPHY          2.526162e+07  4.114078  16239.429126  188123.708738
SHOPPING             2.336117e+07  4.213014  16932.819178  118748.712329
VIDEO_PLAYERS        2.121028e+07  4.019469  16281.253097  204681.946903
TOOLS                1.925128e+07  4.007177   8484.219458  133337.633174
NEWS_AND_MAGAZINES   1.533052e+07  4.143506  12713.563636   38670.259740
SPORTS               1.342222e+07  4.200905  27033.515837  113725.389140
FOOD_AND_DRINK       1.314539e+07  4.109722  22964.622222   46177.527778
SOCIAL               1.280923e+07  4.257692  17728.923077  131664.653846
PERSONALIZATION      1.147858e+07  4.324453  11856.904380  102438.248175
TRAVEL_AND_LOCAL     9.919421e+06  4.011348  25367.004255   31647.326241
WEATHER              9.785930e+06  4.242000  13223.288000   74133.900000
FAMILY               9.071130e+06  4.180330  29682.500330   79913.601320
EDUCATION            8.945477e+06  4.373864  19456.659091   47422.409091
MAPS_AND_NAVIGATION  8.553357e+06  4.008511  17589.055319   38399.521277
ART_AND_DESIGN       6.366002e+06  4.381034  13250.206897   18745.172414
LIFESTYLE            5.853231e+06  4.089963  15880.507063   28508.107807
LIBRARIES_AND_DEMO   5.333050e+06  4.205000  12364.378333   14424.200000
BOOKS_AND_REFERENCE  4.643859e+06  4.322695  14506.587234   23138.234043
HOUSE_AND_HOME       4.629640e+06  4.128000  18102.272000   19833.040000
BUSINESS             4.577851e+06  4.096396  14268.742342   19777.873874
```

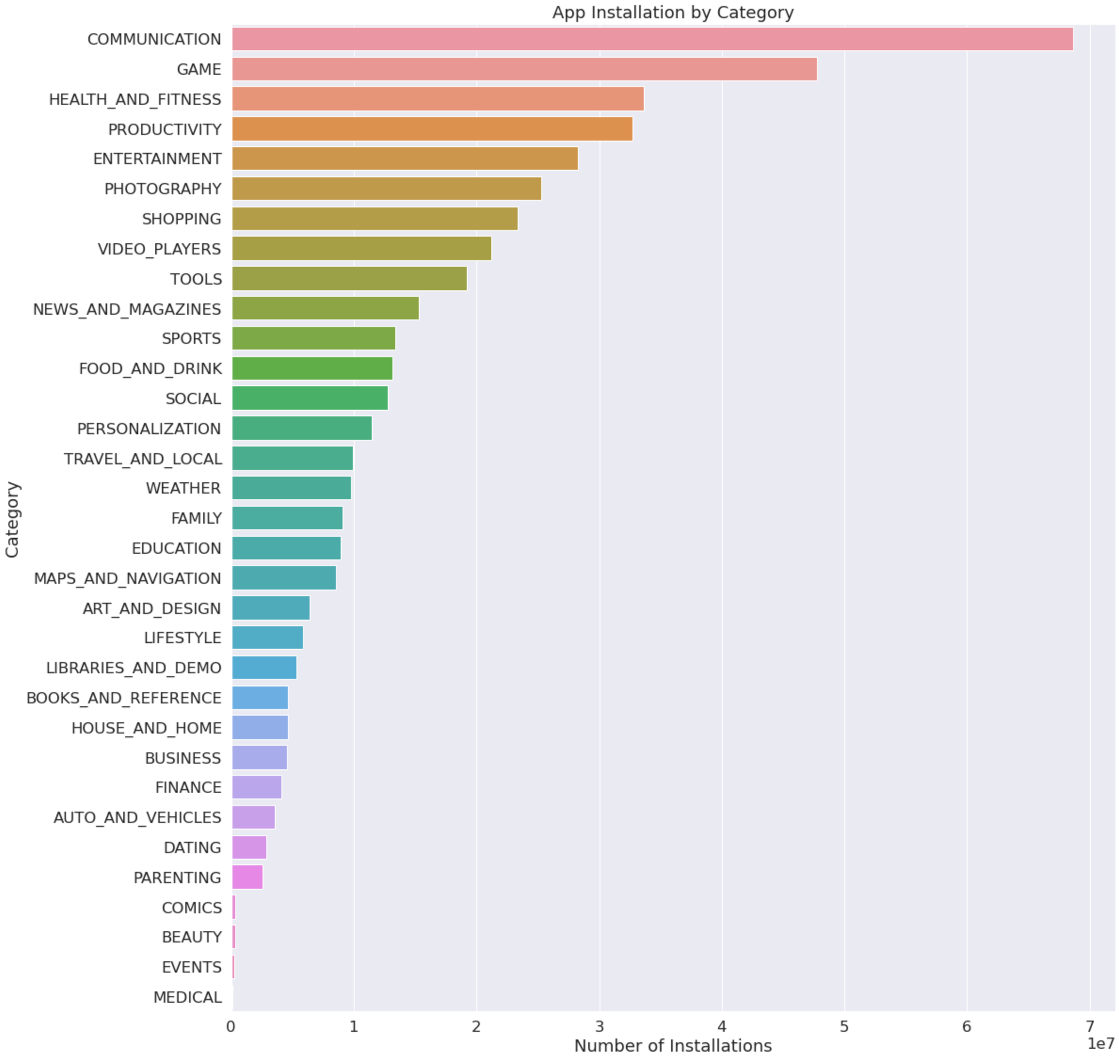
```
FINANCE              4.087548e+06  4.104651  18395.052713   34377.201550
AUTO_AND_VEHICLES    3.551902e+06  4.147619  22058.200000   15750.571429
DATING               2.906495e+06  3.963934  18082.832787   18008.622951
PARENTING            2.581046e+06  4.347727  22097.454545   19999.500000
COMICS               3.708745e+05  4.168085  13789.757447   13048.404255
BEAUTY               3.626000e+05  4.291892  15885.837838    5020.243243
EVENTS               2.802211e+05  4.478947  14778.642105    3321.605263
MEDICAL              1.246673e+05  4.167407  19254.183704    4303.003704

                        Price
Category
COMMUNICATION        0.220794
GAME                 0.325018
HEALTH_AND_FITNESS   0.180209
PRODUCTIVITY         0.236429
ENTERTAINMENT        0.046719
PHOTOGRAPHY          0.322184
SHOPPING             0.037534
VIDEO_PLAYERS        0.008761
TOOLS                0.292823
NEWS_AND_MAGAZINES   0.025844
SPORTS               0.363032
FOOD_AND_DRINK       0.069306
SOCIAL               0.012692
PERSONALIZATION      0.420876
TRAVEL_AND_LOCAL     0.188014
WEATHER              0.468800
FAMILY               1.464310
EDUCATION            0.204091
MAPS_AND_NAVIGATION  0.159149
ART_AND_DESIGN       0.102931
LIFESTYLE            7.261710
LIBRARIES_AND_DEMO   0.000000
BOOKS_AND_REFERENCE  0.148156
HOUSE_AND_HOME       0.000000
BUSINESS             0.231622
FINANCE              9.456860
AUTO_AND_VEHICLES    0.000000
DATING               0.122787
PARENTING            0.113409
COMICS               0.000000
BEAUTY               0.000000
EVENTS               0.000000
MEDICAL              2.166519
```

[58]:

```
plt.figure(figsize= (15, 18))
sns.barplot(y = Category_rate_Install.index, x = Category_rate_Install.Installs)
```

```
plt.title("App Installation by Category")
plt.xlabel("Number of Installations");
```



Number of Reviews for each Category

[59]:
```
Category_rate_Review = Category_rate.sort_values('Reviews', ascending=False)
Category_rate_Review
```

[59]:
```
                     Installs    Rating          Size        Reviews  \
Category
GAME             4.773336e+07  4.236886  43845.124311  628736.355689
COMMUNICATION    6.867151e+07  4.076720  12226.704762  242959.502646
VIDEO_PLAYERS    2.121028e+07  4.019469  16281.253097  204681.946903
PHOTOGRAPHY      2.526162e+07  4.114078  16239.429126  188123.708738
ENTERTAINMENT    2.824938e+07  4.154687  23596.800000  157911.734375
```

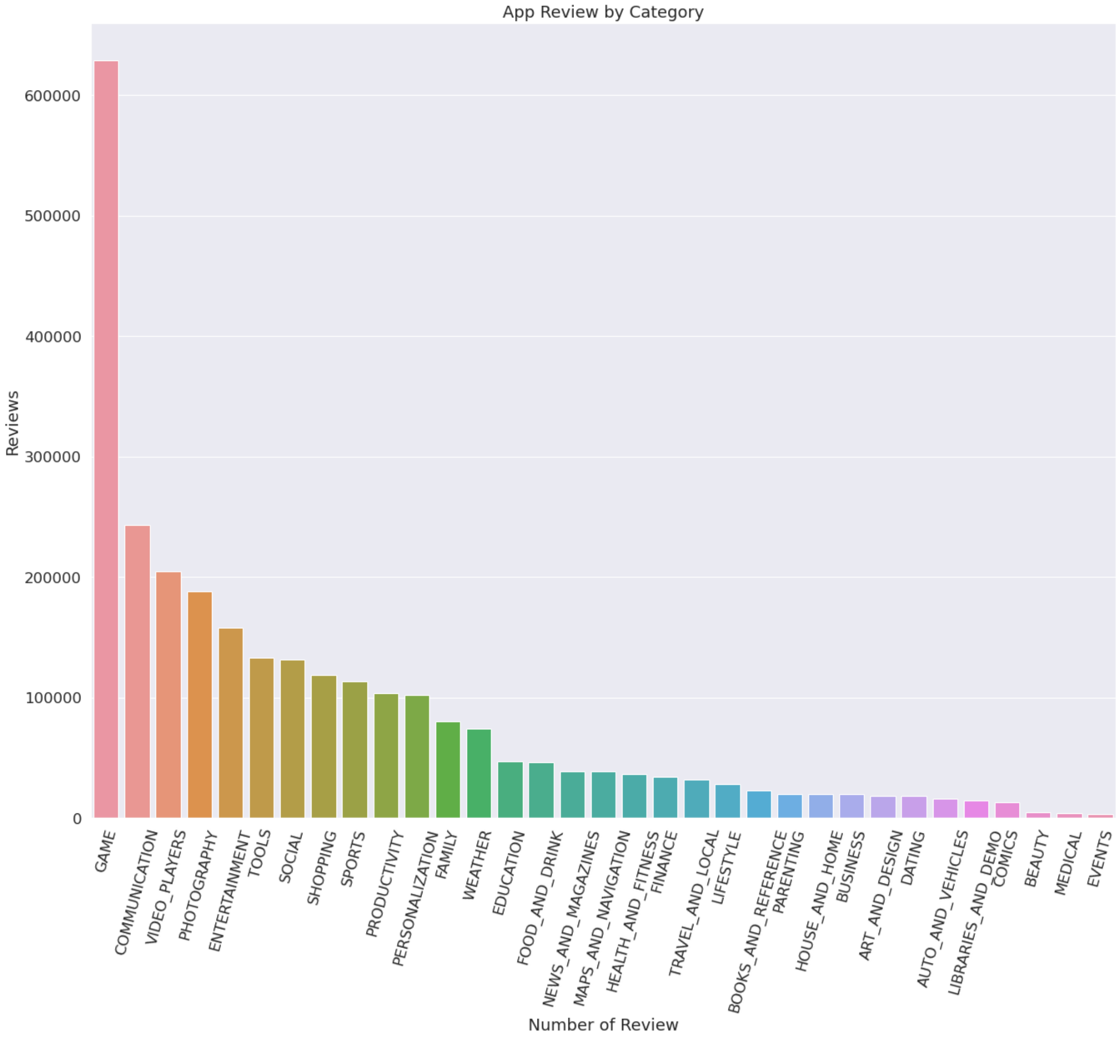
```
TOOLS               1.925128e+07  4.007177   8484.219458  133337.633174
SOCIAL              1.280923e+07  4.257692  17728.923077  131664.653846
SHOPPING            2.336117e+07  4.213014  16932.819178  118748.712329
SPORTS              1.342222e+07  4.200905  27033.515837  113725.389140
PRODUCTIVITY        3.272903e+07  4.131250  12313.932143  103773.040179
PERSONALIZATION     1.147858e+07  4.324453  11856.904380  102438.248175
FAMILY              9.071130e+06  4.180330  29682.500330   79913.601320
WEATHER             9.785930e+06  4.242000  13223.288000   74133.900000
EDUCATION           8.945477e+06  4.373864  19456.659091   47422.409091
FOOD_AND_DRINK      1.314539e+07  4.109722  22964.622222   46177.527778
NEWS_AND_MAGAZINES  1.533052e+07  4.143506  12713.563636   38670.259740
MAPS_AND_NAVIGATION 8.553357e+06  4.008511  17589.055319   38399.521277
HEALTH_AND_FITNESS  3.364637e+07  4.191099  22459.813613   36209.261780
FINANCE             4.087548e+06  4.104651  18395.052713   34377.201550
TRAVEL_AND_LOCAL    9.919421e+06  4.011348  25367.004255   31647.326241
LIFESTYLE           5.853231e+06  4.089963  15880.507063   28508.107807
BOOKS_AND_REFERENCE 4.643859e+06  4.322695  14506.587234   23138.234043
PARENTING           2.581046e+06  4.347727  22097.454545   19999.500000
HOUSE_AND_HOME      4.629640e+06  4.128000  18102.272000   19833.040000
BUSINESS            4.577851e+06  4.096396  14268.742342   19777.873874
ART_AND_DESIGN      6.366002e+06  4.381034  13250.206897   18745.172414
DATING              2.906495e+06  3.963934  18082.832787   18008.622951
AUTO_AND_VEHICLES   3.551902e+06  4.147619  22058.200000   15750.571429
LIBRARIES_AND_DEMO  5.333050e+06  4.205000  12364.378333   14424.200000
COMICS              3.708745e+05  4.168085  13789.757447   13048.404255
BEAUTY              3.626000e+05  4.291892  15885.837838    5020.243243
MEDICAL             1.246673e+05  4.167407  19254.183704    4303.003704
EVENTS              2.802211e+05  4.478947  14778.642105    3321.605263

                       Price
Category
GAME                0.325018
COMMUNICATION       0.220794
VIDEO_PLAYERS       0.008761
PHOTOGRAPHY         0.322184
ENTERTAINMENT       0.046719
TOOLS               0.292823
SOCIAL              0.012692
SHOPPING            0.037534
SPORTS              0.363032
PRODUCTIVITY        0.236429
PERSONALIZATION     0.420876
FAMILY              1.464310
WEATHER             0.468800
EDUCATION           0.204091
FOOD_AND_DRINK      0.069306
NEWS_AND_MAGAZINES  0.025844
```

```
MAPS_AND_NAVIGATION  0.159149
HEALTH_AND_FITNESS   0.180209
FINANCE              9.456860
TRAVEL_AND_LOCAL     0.188014
LIFESTYLE            7.261710
BOOKS_AND_REFERENCE  0.148156
PARENTING            0.113409
HOUSE_AND_HOME       0.000000
BUSINESS             0.231622
ART_AND_DESIGN       0.102931
DATING               0.122787
AUTO_AND_VEHICLES    0.000000
LIBRARIES_AND_DEMO   0.000000
COMICS               0.000000
BEAUTY               0.000000
MEDICAL              2.166519
EVENTS               0.000000
```

[60]:

```
plt.figure(figsize= (18, 15))
sns.barplot(x = Category_rate_Review.index, y = Category_rate_Review.Reviews)
plt.title("App Review by Category")
plt.xlabel("Number of Review")
plt.xticks(rotation = 75);
```



What is the average price of App in each CATEGORY?

[61]:
```
Category_rate_Price = Category_rate.sort_values('Price', ascending=False)
Category_rate_Price
```

[61]:
```
                      Installs    Rating          Size        Reviews  \
Category
FINANCE           4.087548e+06  4.104651  18395.052713   34377.201550
LIFESTYLE         5.853231e+06  4.089963  15880.507063   28508.107807
MEDICAL           1.246673e+05  4.167407  19254.183704    4303.003704
FAMILY            9.071130e+06  4.180330  29682.500330   79913.601320
WEATHER           9.785930e+06  4.242000  13223.288000   74133.900000
PERSONALIZATION   1.147858e+07  4.324453  11856.904380  102438.248175
SPORTS            1.342222e+07  4.200905  27033.515837  113725.389140
GAME              4.773336e+07  4.236886  43845.124311  628736.355689
```

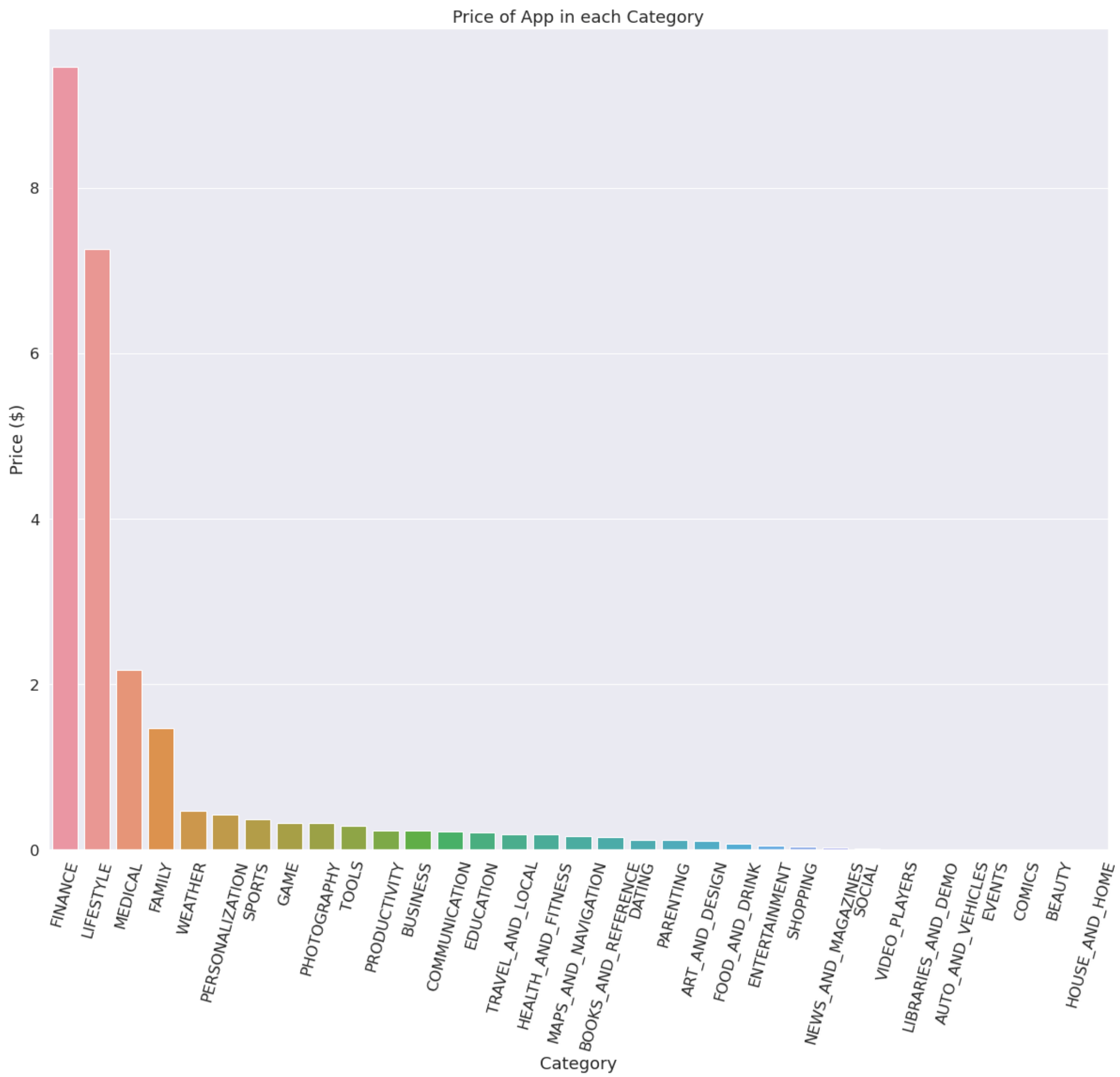
```
PHOTOGRAPHY          2.526162e+07  4.114078  16239.429126  188123.708738
TOOLS                1.925128e+07  4.007177   8484.219458  133337.633174
PRODUCTIVITY         3.272903e+07  4.131250  12313.932143  103773.040179
BUSINESS             4.577851e+06  4.096396  14268.742342   19777.873874
COMMUNICATION        6.867151e+07  4.076720  12226.704762  242959.502646
EDUCATION            8.945477e+06  4.373864  19456.659091   47422.409091
TRAVEL_AND_LOCAL     9.919421e+06  4.011348  25367.004255   31647.326241
HEALTH_AND_FITNESS   3.364637e+07  4.191099  22459.813613   36209.261780
MAPS_AND_NAVIGATION  8.553357e+06  4.008511  17589.055319   38399.521277
BOOKS_AND_REFERENCE  4.643859e+06  4.322695  14506.587234   23138.234043
DATING               2.906495e+06  3.963934  18082.832787   18008.622951
PARENTING            2.581046e+06  4.347727  22097.454545   19999.500000
ART_AND_DESIGN       6.366002e+06  4.381034  13250.206897   18745.172414
FOOD_AND_DRINK       1.314539e+07  4.109722  22964.622222   46177.527778
ENTERTAINMENT        2.824938e+07  4.154687  23596.800000  157911.734375
SHOPPING             2.336117e+07  4.213014  16932.819178  118748.712329
NEWS_AND_MAGAZINES   1.533052e+07  4.143506  12713.563636   38670.259740
SOCIAL               1.280923e+07  4.257692  17728.923077  131664.653846
VIDEO_PLAYERS        2.121028e+07  4.019469  16281.253097  204681.946903
LIBRARIES_AND_DEMO   5.333050e+06  4.205000  12364.378333   14424.200000
AUTO_AND_VEHICLES    3.551902e+06  4.147619  22058.200000   15750.571429
EVENTS               2.802211e+05  4.478947  14778.642105    3321.605263
COMICS               3.708745e+05  4.168085  13789.757447   13048.404255
BEAUTY               3.626000e+05  4.291892  15885.837838    5020.243243
HOUSE_AND_HOME       4.629640e+06  4.128000  18102.272000   19833.040000

                        Price
Category
FINANCE              9.456860
LIFESTYLE            7.261710
MEDICAL              2.166519
FAMILY               1.464310
WEATHER              0.468800
PERSONALIZATION      0.420876
SPORTS               0.363032
GAME                 0.325018
PHOTOGRAPHY          0.322184
TOOLS                0.292823
PRODUCTIVITY         0.236429
BUSINESS             0.231622
COMMUNICATION        0.220794
EDUCATION            0.204091
TRAVEL_AND_LOCAL     0.188014
HEALTH_AND_FITNESS   0.180209
MAPS_AND_NAVIGATION  0.159149
BOOKS_AND_REFERENCE  0.148156
DATING               0.122787
```

```
PARENTING            0.113409
ART_AND_DESIGN       0.102931
FOOD_AND_DRINK       0.069306
ENTERTAINMENT        0.046719
SHOPPING             0.037534
NEWS_AND_MAGAZINES   0.025844
SOCIAL               0.012692
VIDEO_PLAYERS        0.008761
LIBRARIES_AND_DEMO   0.000000
AUTO_AND_VEHICLES    0.000000
EVENTS               0.000000
COMICS               0.000000
BEAUTY               0.000000
HOUSE_AND_HOME       0.000000
```

[62]:

```
plt.figure(figsize= (18, 15))
sns.barplot(x = Category_rate_Price.index, y = Category_rate_Price.Price)
plt.title("Price of App in each Category")
plt.ylabel("Price ($)")
plt.xticks(rotation = 75);
```



What is the average size of app in each Category?

[63]:
```
Category_rate_Size = Category_rate.sort_values('Size', ascending=False)
Category_rate_Size
```

[63]:
```
                        Installs    Rating          Size        Reviews  \
Category
GAME                4.773336e+07  4.236886  43845.124311  628736.355689
FAMILY              9.071130e+06  4.180330  29682.500330   79913.601320
SPORTS              1.342222e+07  4.200905  27033.515837  113725.389140
TRAVEL_AND_LOCAL    9.919421e+06  4.011348  25367.004255   31647.326241
ENTERTAINMENT       2.824938e+07  4.154687  23596.800000  157911.734375
FOOD_AND_DRINK      1.314539e+07  4.109722  22964.622222   46177.527778
HEALTH_AND_FITNESS  3.364637e+07  4.191099  22459.813613   36209.261780
```

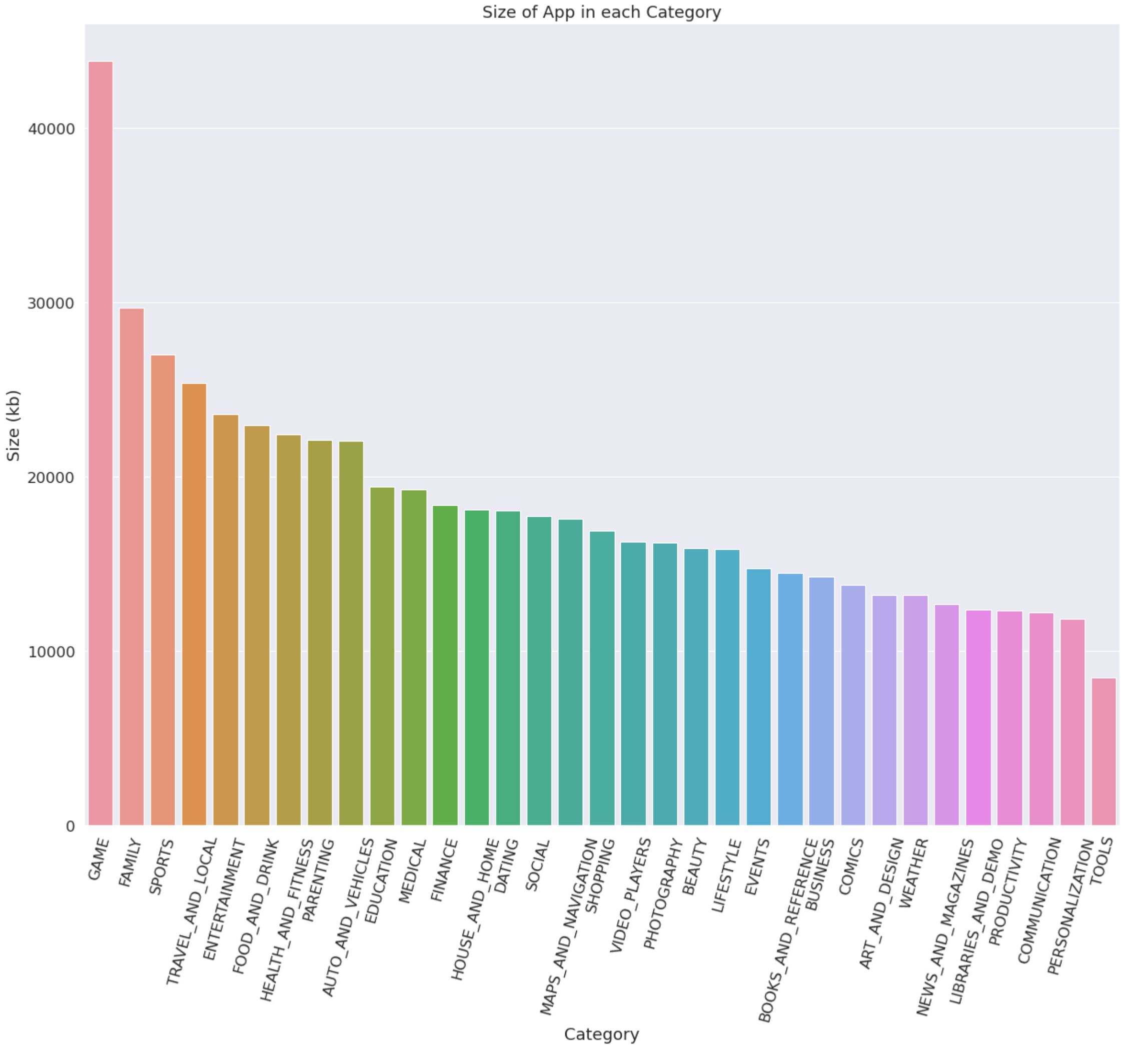
```
PARENTING            2.581046e+06  4.347727  22097.454545   19999.500000
AUTO_AND_VEHICLES    3.551902e+06  4.147619  22058.200000   15750.571429
EDUCATION            8.945477e+06  4.373864  19456.659091   47422.409091
MEDICAL              1.246673e+05  4.167407  19254.183704    4303.003704
FINANCE              4.087548e+06  4.104651  18395.052713   34377.201550
HOUSE_AND_HOME       4.629640e+06  4.128000  18102.272000   19833.040000
DATING               2.906495e+06  3.963934  18082.832787   18008.622951
SOCIAL               1.280923e+07  4.257692  17728.923077  131664.653846
MAPS_AND_NAVIGATION  8.553357e+06  4.008511  17589.055319   38399.521277
SHOPPING             2.336117e+07  4.213014  16932.819178  118748.712329
VIDEO_PLAYERS        2.121028e+07  4.019469  16281.253097  204681.946903
PHOTOGRAPHY          2.526162e+07  4.114078  16239.429126  188123.708738
BEAUTY               3.626000e+05  4.291892  15885.837838    5020.243243
LIFESTYLE            5.853231e+06  4.089963  15880.507063   28508.107807
EVENTS               2.802211e+05  4.478947  14778.642105    3321.605263
BOOKS_AND_REFERENCE  4.643859e+06  4.322695  14506.587234   23138.234043
BUSINESS             4.577851e+06  4.096396  14268.742342   19777.873874
COMICS               3.708745e+05  4.168085  13789.757447   13048.404255
ART_AND_DESIGN       6.366002e+06  4.381034  13250.206897   18745.172414
WEATHER              9.785930e+06  4.242000  13223.288000   74133.900000
NEWS_AND_MAGAZINES   1.533052e+07  4.143506  12713.563636   38670.259740
LIBRARIES_AND_DEMO   5.333050e+06  4.205000  12364.378333   14424.200000
PRODUCTIVITY         3.272903e+07  4.131250  12313.932143  103773.040179
COMMUNICATION        6.867151e+07  4.076720  12226.704762  242959.502646
PERSONALIZATION      1.147858e+07  4.324453  11856.904380  102438.248175
TOOLS                1.925128e+07  4.007177   8484.219458  133337.633174

                        Price
Category
GAME                 0.325018
FAMILY               1.464310
SPORTS               0.363032
TRAVEL_AND_LOCAL     0.188014
ENTERTAINMENT        0.046719
FOOD_AND_DRINK       0.069306
HEALTH_AND_FITNESS   0.180209
PARENTING            0.113409
AUTO_AND_VEHICLES    0.000000
EDUCATION            0.204091
MEDICAL              2.166519
FINANCE              9.456860
HOUSE_AND_HOME       0.000000
DATING               0.122787
SOCIAL               0.012692
MAPS_AND_NAVIGATION  0.159149
SHOPPING             0.037534
VIDEO_PLAYERS        0.008761
```

```
PHOTOGRAPHY          0.322184
BEAUTY               0.000000
LIFESTYLE            7.261710
EVENTS               0.000000
BOOKS_AND_REFERENCE  0.148156
BUSINESS             0.231622
COMICS               0.000000
ART_AND_DESIGN       0.102931
WEATHER              0.468800
NEWS_AND_MAGAZINES   0.025844
LIBRARIES_AND_DEMO   0.000000
PRODUCTIVITY         0.236429
COMMUNICATION        0.220794
PERSONALIZATION      0.420876
TOOLS                0.292823
```

[64]:

```
plt.figure(figsize= (18, 15))
sns.barplot(x = Category_rate_Size.index, y = Category_rate_Size.Size)
plt.title("Size of App in each Category")
plt.ylabel("Size (kb)")
plt.xticks(rotation = 75);
```



What is the average ratings of each category?

[65]:

```
Category_ratings = Category_rate.sort_values('Rating', ascending=False)
Category_ratings
```

[65]:

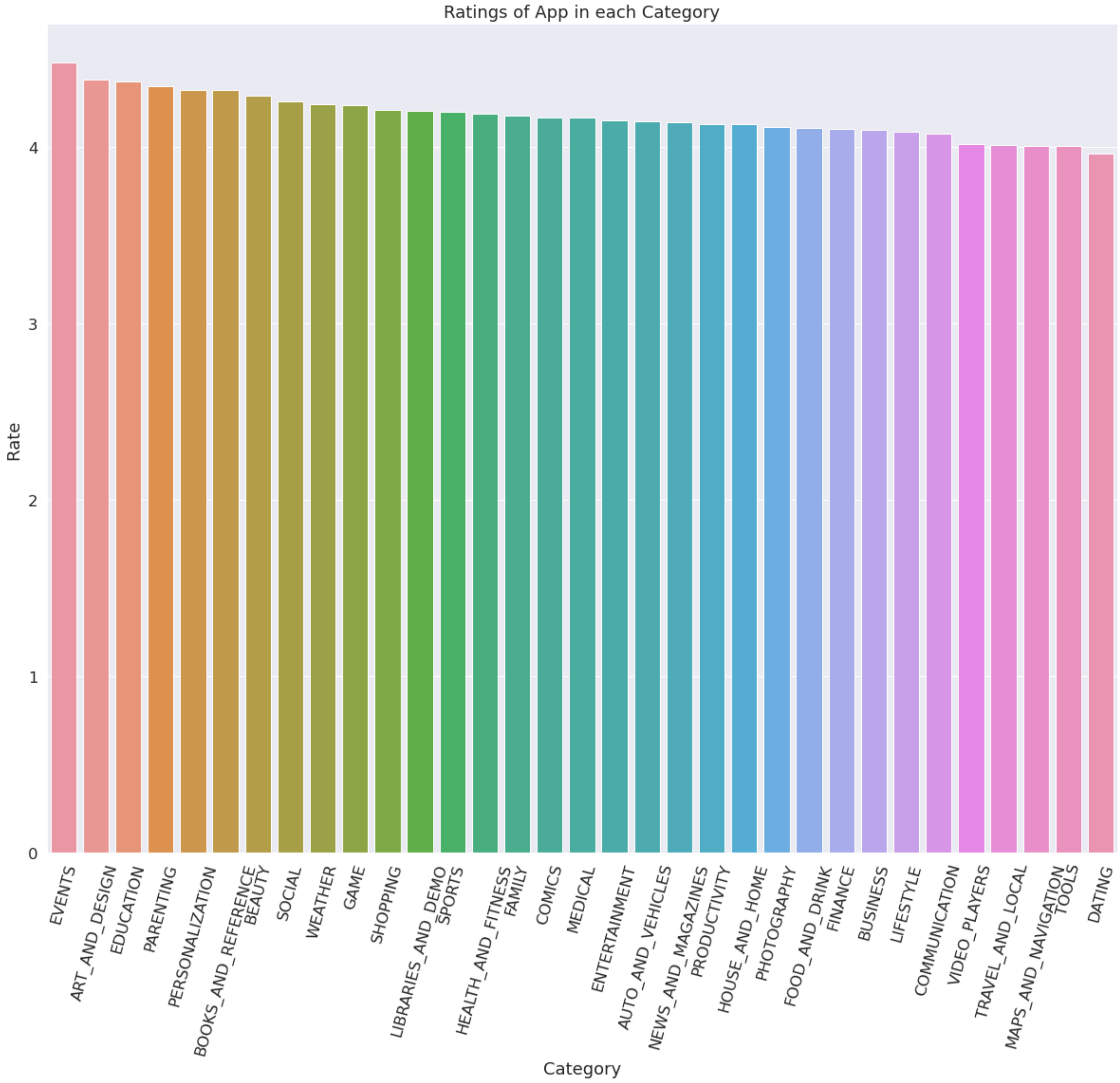
```
                         Installs    Rating          Size        Reviews  \
Category
EVENTS               2.802211e+05  4.478947  14778.642105    3321.605263
ART_AND_DESIGN       6.366002e+06  4.381034  13250.206897   18745.172414
EDUCATION            8.945477e+06  4.373864  19456.659091   47422.409091
PARENTING            2.581046e+06  4.347727  22097.454545   19999.500000
PERSONALIZATION      1.147858e+07  4.324453  11856.904380  102438.248175
BOOKS_AND_REFERENCE  4.643859e+06  4.322695  14506.587234   23138.234043
BEAUTY               3.626000e+05  4.291892  15885.837838    5020.243243
SOCIAL               1.280923e+07  4.257692  17728.923077  131664.653846
```

```
WEATHER              9.785930e+06  4.242000  13223.288000   74133.900000
GAME                 4.773336e+07  4.236886  43845.124311  628736.355689
SHOPPING             2.336117e+07  4.213014  16932.819178  118748.712329
LIBRARIES_AND_DEMO   5.333050e+06  4.205000  12364.378333   14424.200000
SPORTS               1.342222e+07  4.200905  27033.515837  113725.389140
HEALTH_AND_FITNESS   3.364637e+07  4.191099  22459.813613   36209.261780
FAMILY               9.071130e+06  4.180330  29682.500330   79913.601320
COMICS               3.708745e+05  4.168085  13789.757447   13048.404255
MEDICAL              1.246673e+05  4.167407  19254.183704    4303.003704
ENTERTAINMENT        2.824938e+07  4.154687  23596.800000  157911.734375
AUTO_AND_VEHICLES    3.551902e+06  4.147619  22058.200000   15750.571429
NEWS_AND_MAGAZINES   1.533052e+07  4.143506  12713.563636   38670.259740
PRODUCTIVITY         3.272903e+07  4.131250  12313.932143  103773.040179
HOUSE_AND_HOME       4.629640e+06  4.128000  18102.272000   19833.040000
PHOTOGRAPHY          2.526162e+07  4.114078  16239.429126  188123.708738
FOOD_AND_DRINK       1.314539e+07  4.109722  22964.622222   46177.527778
FINANCE              4.087548e+06  4.104651  18395.052713   34377.201550
BUSINESS             4.577851e+06  4.096396  14268.742342   19777.873874
LIFESTYLE            5.853231e+06  4.089963  15880.507063   28508.107807
COMMUNICATION        6.867151e+07  4.076720  12226.704762  242959.502646
VIDEO_PLAYERS        2.121028e+07  4.019469  16281.253097  204681.946903
TRAVEL_AND_LOCAL     9.919421e+06  4.011348  25367.004255   31647.326241
MAPS_AND_NAVIGATION  8.553357e+06  4.008511  17589.055319   38399.521277
TOOLS                1.925128e+07  4.007177   8484.219458  133337.633174
DATING               2.906495e+06  3.963934  18082.832787   18008.622951

                        Price
Category
EVENTS               0.000000
ART_AND_DESIGN       0.102931
EDUCATION            0.204091
PARENTING            0.113409
PERSONALIZATION      0.420876
BOOKS_AND_REFERENCE  0.148156
BEAUTY               0.000000
SOCIAL               0.012692
WEATHER              0.468800
GAME                 0.325018
SHOPPING             0.037534
LIBRARIES_AND_DEMO   0.000000
SPORTS               0.363032
HEALTH_AND_FITNESS   0.180209
FAMILY               1.464310
COMICS               0.000000
MEDICAL              2.166519
ENTERTAINMENT        0.046719
AUTO_AND_VEHICLES    0.000000
```

```
NEWS_AND_MAGAZINES   0.025844
PRODUCTIVITY         0.236429
HOUSE_AND_HOME       0.000000
PHOTOGRAPHY          0.322184
FOOD_AND_DRINK       0.069306
FINANCE              9.456860
BUSINESS             0.231622
LIFESTYLE            7.261710
COMMUNICATION        0.220794
VIDEO_PLAYERS        0.008761
TRAVEL_AND_LOCAL     0.188014
MAPS_AND_NAVIGATION  0.159149
TOOLS                0.292823
DATING               0.122787
```

[66]:
```
plt.figure(figsize= (18, 15))
sns.barplot(x = Category_ratings.index, y = Category_ratings.Rating)
plt.title("Ratings of App in each Category")
plt.ylabel("Rate ")
plt.xticks(rotation = 75);
```


Ratings of App in each Category
Rate
0
1
2
3
4
Category
EVENTS
ART_AND_DESIGN
EDUCATION
PARENTING
PERSONALIZATION
BOOKS_AND_REFERENCE
BEAUTY
SOCIAL
WEATHER
GAME
SHOPPING
LIBRARIES_AND_DEMO
SPORTS
HEALTH_AND_FITNESS
FAMILY
COMICS
MEDICAL
ENTERTAINMENT
AUTO_AND_VEHICLES
NEWS_AND_MAGAZINES
PRODUCTIVITY
HOUSE_AND_HOME
PHOTOGRAPHY
FOOD_AND_DRINK
FINANCE
BUSINESS
LIFESTYLE
COMMUNICATION
VIDEO_PLAYERS
TRAVEL_AND_LOCAL
MAPS_AND_NAVIGATION
TOOLS
DATING