In [ ]:

```
Data Analysis Project  - Parag Y
```

"Obective: Make a model to predict the app rating, with other information about the app provided"

In [191]:

```python
#importing Libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

1.

In [192]:

```python
#1. Load the data files using Pandas

data = pd.read_csv('googleplaystore.csv')
```

Knowing the Data

In [193]:

```python
data.head()
```

Out[193]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [194]:

```python
data.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
```

In [195]:

```python
data.shape
```

Out[195]:

```
(10841, 13)
```

2.

In [196]:

```
##2. Checking for null values count by each column

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

Out[196]:

```
App               False
Category          False
Rating             True
Reviews           False
Size              False
Installs          False
Type               True
Price             False
Content Rating     True
Genres            False
Last Updated      False
Current Ver        True
Android Ver        True
dtype: bool
```

In [197]:

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

Out[197]:

```
App                  0
Category             0
Rating            1474
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
```

Data Wrangling

3.

In [198]:

```
#3. Droping the records with null in any of the column,
#Since the question demands of removing all the null items we will not go by

data = data.dropna()
```

In [199]:

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

Out[199]:

```
App               False
Category          False
Rating            False
Reviews           False
Size              False
Installs          False
Type              False
Price             False
Content Rating    False
Genres            False
Last Updated      False
Current Ver       False
Android Ver       False
dtype: bool
```

In [200]:

```
data.shape
```

Out[200]:

```
(9360, 13)
```

4(I).

As the model do not understand categorical variable so before moving towards the visualization all categorical Data types must be converted to numeric on which the analysis is to be done

In [201]:

```python
data["Size"] = [ float(i.split('M')[0]) if 'M' in i else float(0) for i in data["Size"]  ]
```

In [202]:

```python
data.head()
```

Out[202]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [203]:

```python
data["Size"] = 1000 * data["Size"]
```

In [204]:

```python
data
```

Out[204]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19000.0	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14000.0	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8700.0	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25000.0	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2800.0	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up
...	...	...	...	...	...	...	...	...	...	...	...	...	...
10834	FR Calculator	FAMILY	4.0	7	2600.0	500+	Free	0	Everyone	Education	June 18, 2017	1.0.0	4.1 and up
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53000.0	5,000+	Free	0	Everyone	Education	July 25, 2017	1.48	4.1 and up
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3600.0	100+	Free	0	Everyone	Education	July 6, 2018	1.0	4.1 and up
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	0.0	1,000+	Free	0	Mature 17+	Books & Reference	January 19, 2015	Varies with device	Varies with device
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19000.0	10,000,000+	Free	0	Everyone	Lifestyle	July 25, 2018	Varies with device	Varies with device

9360 rows × 13 columns

4(II).

In [205]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   object 
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   object 
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   object 
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(2), object(11)
memory usage: 1023.8+ KB
```

In [206]:

```
data["Reviews"] = data["Reviews"].astype(float)
```

In [207]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   float64
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   object 
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   object 
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(3), object(10)
memory usage: 1023.8+ KB
```

4(III).

In [208]:

```
data["Installs"] = [ float(i.replace('+','').replace(',', '')) if '+' in i or ',' in i else float(0) for i in data["Instal
```

In [209]:

```
data.head()
```

Out[209]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000.0	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000.0	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000.0	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000.0	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000.0	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [210]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   float64
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   float64
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   object 
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(4), object(9)
memory usage: 1023.8+ KB
```

In [211]:
```
data["Installs"] = data["Installs"].astype(int)
```

In [212]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   float64
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   int32  
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   object 
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(3), int32(1), object(9)
memory usage: 987.2+ KB
```

4(IV).

In [213]:
```
data['Price'] = [ float(i.split('$')[1]) if '$' in i else float(0) for i in data['Price'] ]
```

In [214]:
```
data.head()
```

Out[214]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000	Free	0.0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000	Free	0.0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000	Free	0.0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000	Free	0.0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [215]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   float64
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   int32  
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   float64
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(4), int32(1), object(8)
memory usage: 987.2+ KB
```

In [216]:

```
data["Price"] = data["Price"].astype(int)
```

In [217]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9360 non-null   object 
 1   Category        9360 non-null   object 
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   float64
 4   Size            9360 non-null   float64
 5   Installs        9360 non-null   int32  
 6   Type            9360 non-null   object 
 7   Price           9360 non-null   int32  
 8   Content Rating  9360 non-null   object 
 9   Genres          9360 non-null   object 
 10  Last Updated    9360 non-null   object 
 11  Current Ver     9360 non-null   object 
 12  Android Ver     9360 non-null   object 
dtypes: float64(3), int32(2), object(8)
memory usage: 950.6+ KB
```

4(V-A).

In [218]:

```
data.shape
```

Out[218]:

```
(9360, 13)
```

In [219]:

```
data.drop(data[(data['Reviews'] < 1) & (data['Reviews'] > 5 )].index, inplace = True)
```

In [220]:

```
data.shape
```

Out[220]:

```
(9360, 13)
```

4(V-B).

In [221]:

```
data.shape
```

Out[221]:

```
(9360, 13)
```

In [222]:

```
data.drop(data[data['Installs'] < data['Reviews'] ].index, inplace = True)
```

In [223]:

```
data.shape
```

Out[223]:

```
(9353, 13)
```

4(V-C).

```
In [224]: data.shape
```

```
Out[224]: (9353, 13)
```

```
In [225]: data.drop(data[(data['Type'] =='Free') & (data['Price'] > 0 )].index, inplace = True)
```

```
In [226]: data.shape
```

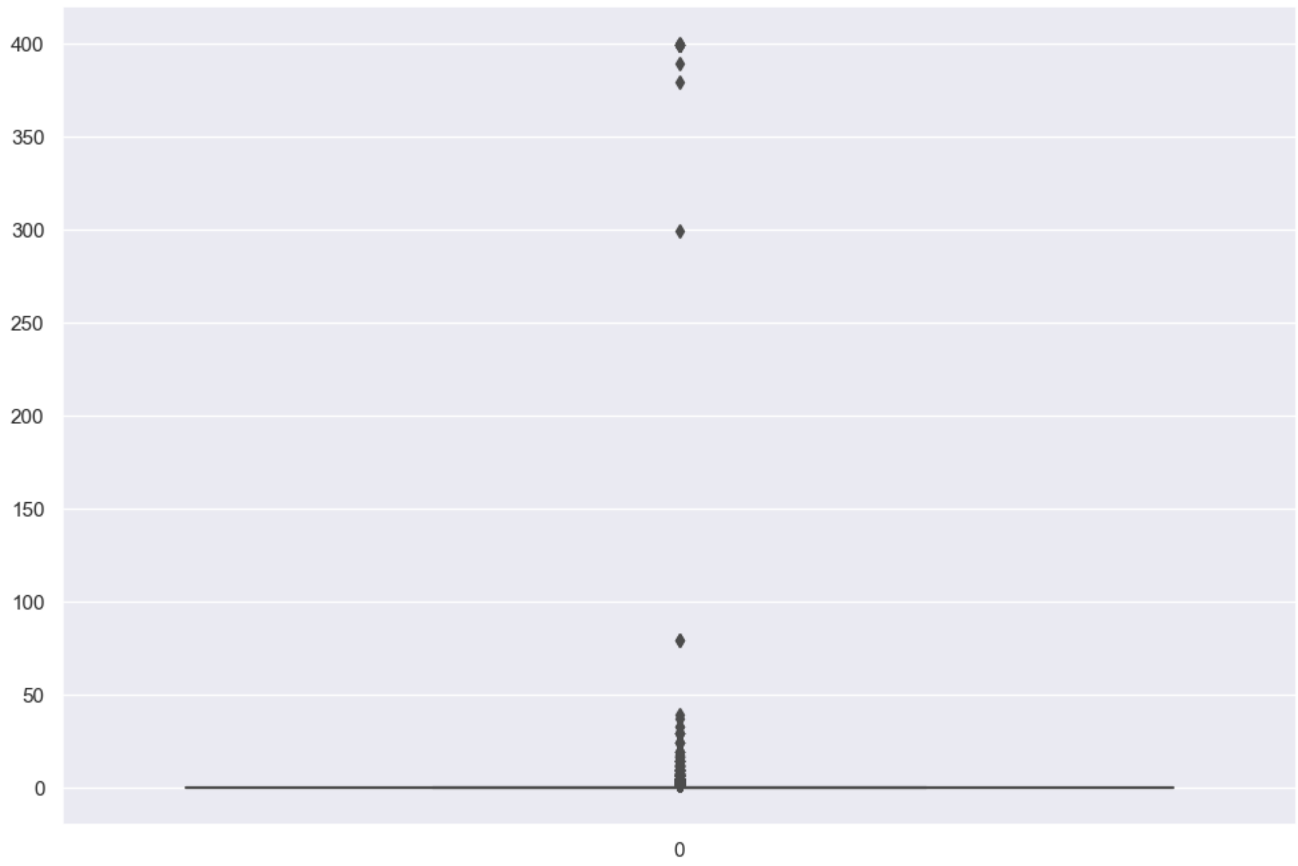
```
Out[226]: (9353, 13)
```

5(I).

```
In [227]: sns.set(rc={'figure.figsize':(12,8)})
```

```
In [228]: sns.boxplot(data['Price'])
```

```
Out[228]: <Axes: >
```



indeed there are some outliers in the Price column,

i.e., there are some apps whose price is more than usual apps on the Googleplaystore

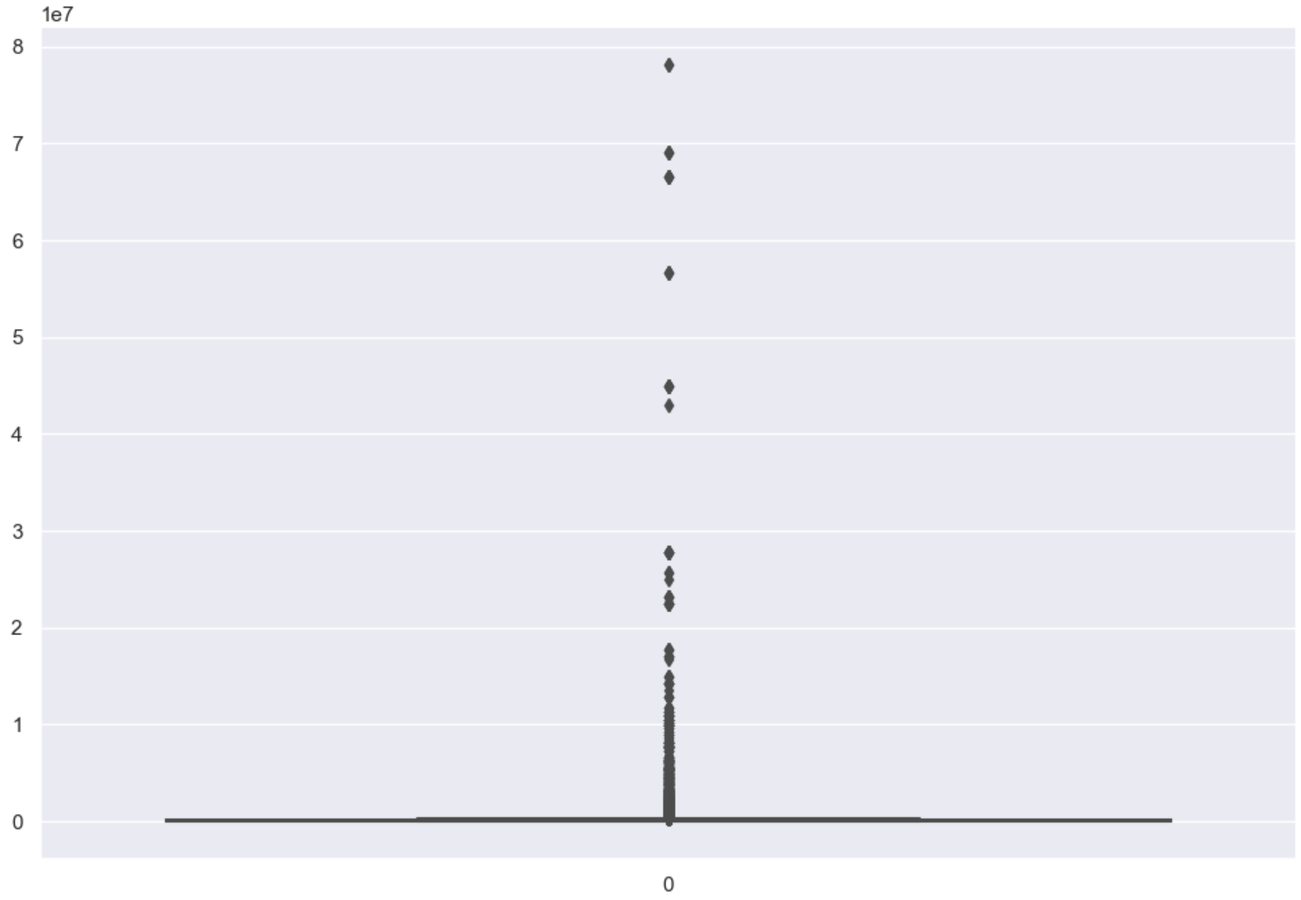
5(II).

In [229]:

```
sns.boxplot(data['Reviews'])
```

Out[229]:

```
<Axes: >
```



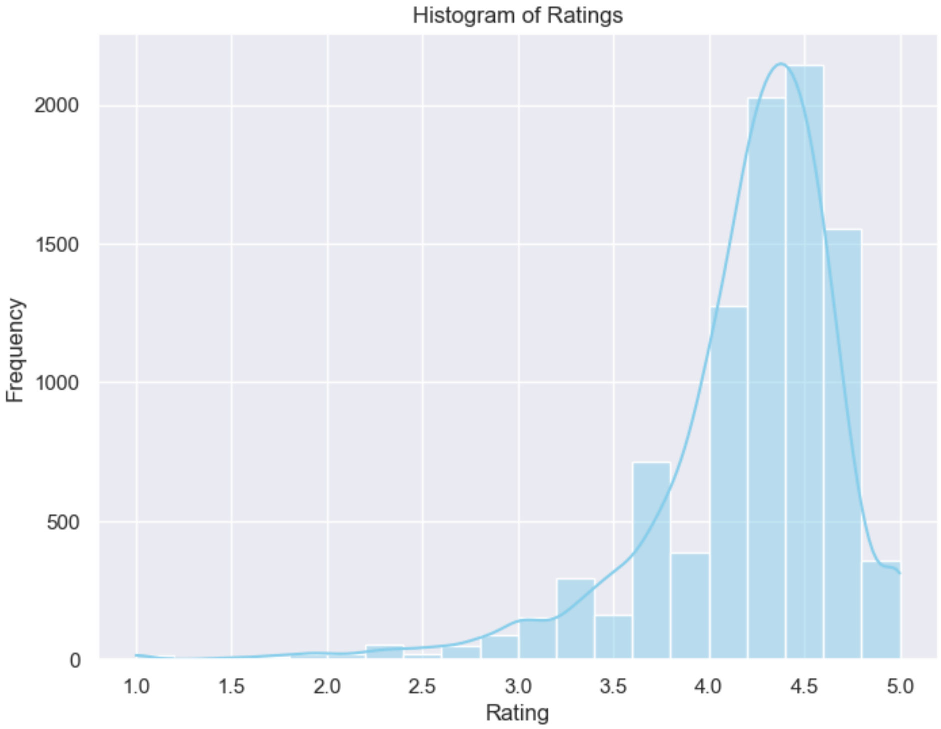
Indeed there are some apps that have very high number of Reviews

5(III).

In [230]:

```python
sns.set(rc={'figure.figsize':(8,6)})

# Create a histogram of the 'Rating' column using Seaborn and Matplotlib
sns.histplot(data['Rating'], bins=20, kde=True, color='skyblue')
plt.title('Histogram of Ratings')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```



There is a Negative skewness(left- skewed)

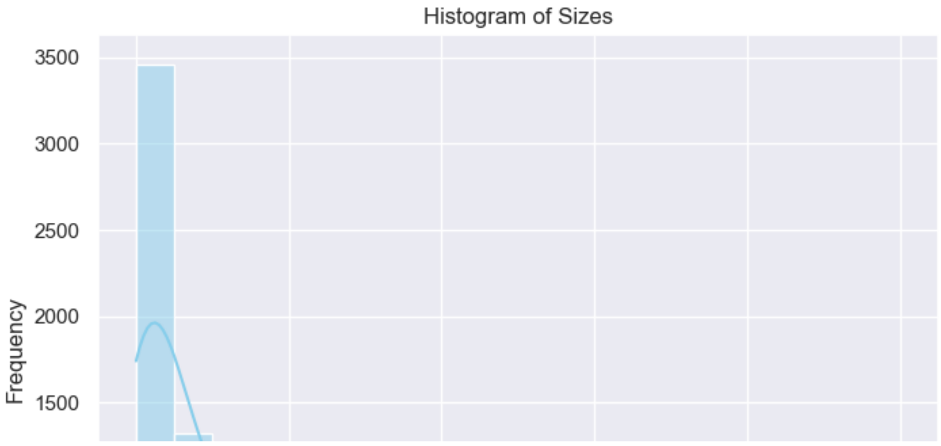
some apps seem to have higher Ratings than usual

5(IV).

In [231]:

```python
sns.set(rc={'figure.figsize':(8,6)})

# Create a histogram of the 'Size' column using Seaborn and Matplotlib
sns.histplot(data['Size'], bins=20, kde=True, color='skyblue')
plt.title('Histogram of Sizes')
plt.xlabel('Size')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```



positive skewness Right Skewed

Handling outliers

6(I).

As per the above observation of plots, there seems to be some outliers in the Price & Reviews column

In the Installs column as well

In [232]:

```
##I) price of $200 and above for an application is expected to be very high

more = data.apply(lambda x : True
            if x['Price'] > 200 else False, axis = 1)
```

In [233]:

```
more_count = len(more[more == True].index)
```

In [234]:

```
data.shape
```

Out[234]:

```
(9353, 13)
```

In [235]:

```
##Dropping the Junk apps
data.drop(data[data['Price'] > 200].index, inplace = True)
```

In [236]:

```
data.shape
```

Out[236]:

```
(9338, 13)
```

6(II).

In [237]:

```
#II) Very few apps have very high no. of Reviews
##Dropping the Star apps as these will skew the analysis,

data.drop(data[data['Reviews'] > 2000000].index, inplace = True)
```

In [238]:

```
data.shape
```

Out[238]:

```
(8885, 13)
```

6(III).

In [239]:

```
##III) Find out the Percentiles of Installs and decide a threshold as cutoff for outlier

data.quantile([.1, .25, .5, .70, .90, .95, .99], axis = 0)
```

```
C:\Users\Parag\AppData\Local\Temp\ipykernel_14540\2378969602.py:3: FutureWarning: The default value of numeric_only in DataFr
ame.quantile is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of
numeric_only to silence this warning.
  data.quantile([.1, .25, .5, .70, .90, .95, .99], axis = 0)
```

Out[239]:

	Rating	Reviews	Size	Installs	Price
0.10	3.5	18.00	0.0	1000.0	0.0
0.25	4.0	159.00	2600.0	10000.0	0.0
0.50	4.3	4290.00	9500.0	500000.0	0.0
0.70	4.5	35930.40	23000.0	1000000.0	0.0
0.90	4.7	296771.00	50000.0	10000000.0	0.0
0.95	4.8	637298.00	68000.0	10000000.0	1.0
0.99	5.0	1462800.88	95000.0	100000000.0	7.0

In [240]:

```
# dropping more than 10000000 Installs value
data.drop(data[data['Installs'] > 10000000].index, inplace = True)
```

In [241]:
```
data.shape
```

Out[241]:
```
(8496, 13)
```

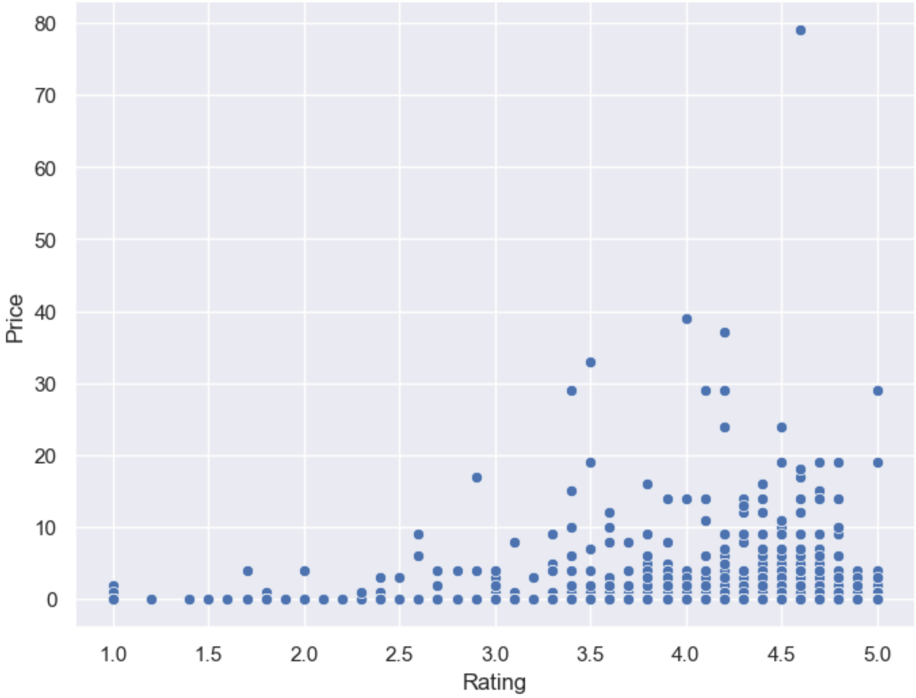
Bivariate Analysis

7(I).

In [242]:
```
##1) Scatter plot/jointplot for Rating Vs. Price

sns.scatterplot(x='Rating',y='Price',data=data)
```

Out[242]:
```
<Axes: xlabel='Rating', ylabel='Price'>
```



That states the paid apps have the highest of Ratings

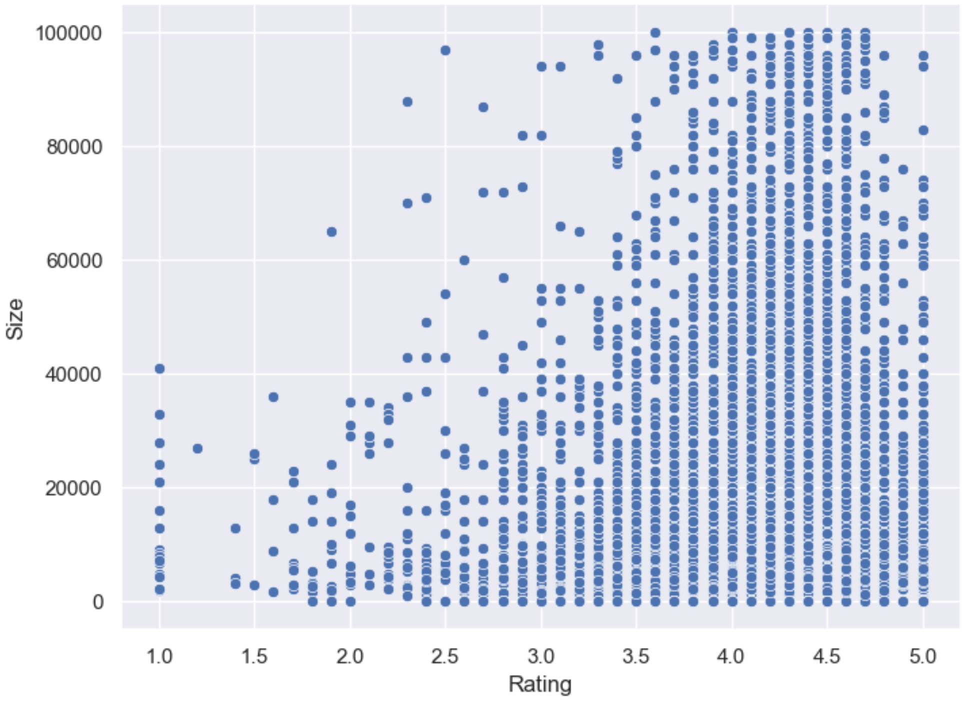
In [243]:

```
#2) Scatterplot/jointplot for Rating Vs. Size

sns.scatterplot(x='Rating',y='Size',data=data)
```

Out[243]:

```
<Axes: xlabel='Rating', ylabel='Size'>
```

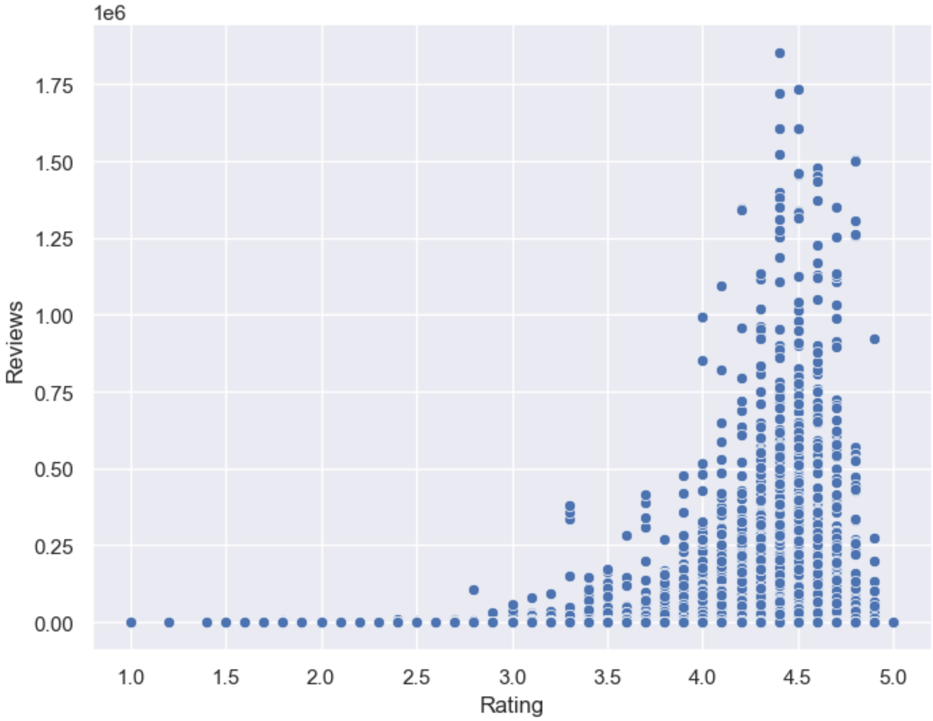


Yes it is clear that heavier apps are rated better.

7(III).

In [244]:
```
##3) Scatterplot for Ratings Vs. Reviews
sns.scatterplot(x='Rating',y='Reviews',data=data)
```

Out[244]:
```
<Axes: xlabel='Rating', ylabel='Reviews'>
```



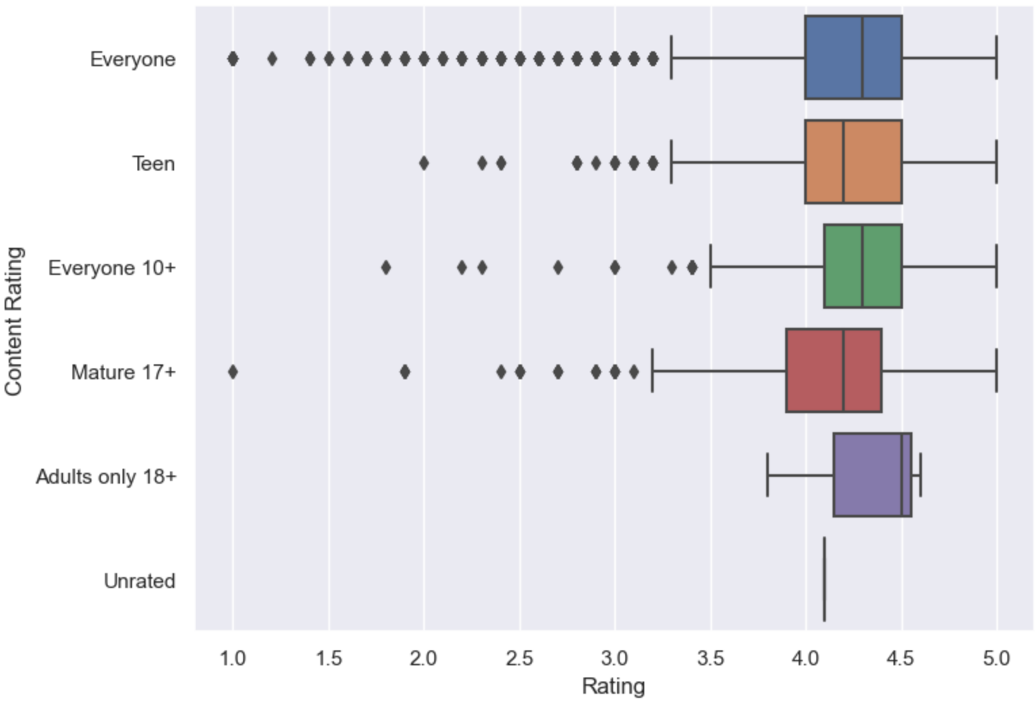
The plot shows a positive linear relationship between Ratings and Reviews. More reviews mean better ratings indeed

7(IV).

In [245]:
```
#4) Boxplot for Ratings Vs. Content Rating

sns.boxplot(x="Rating", y="Content Rating", data=data)
```

Out[245]:
```
<Axes: xlabel='Rating', ylabel='Content Rating'>
```



The above plot shows the apps for Everyone is worst rated as it contain the highest number of outliers followed by apps for Mature 17+ and Everyone 10+ along with Teen. The catergory Adults only 18+ is rated better and falls under most liked type

7(V)

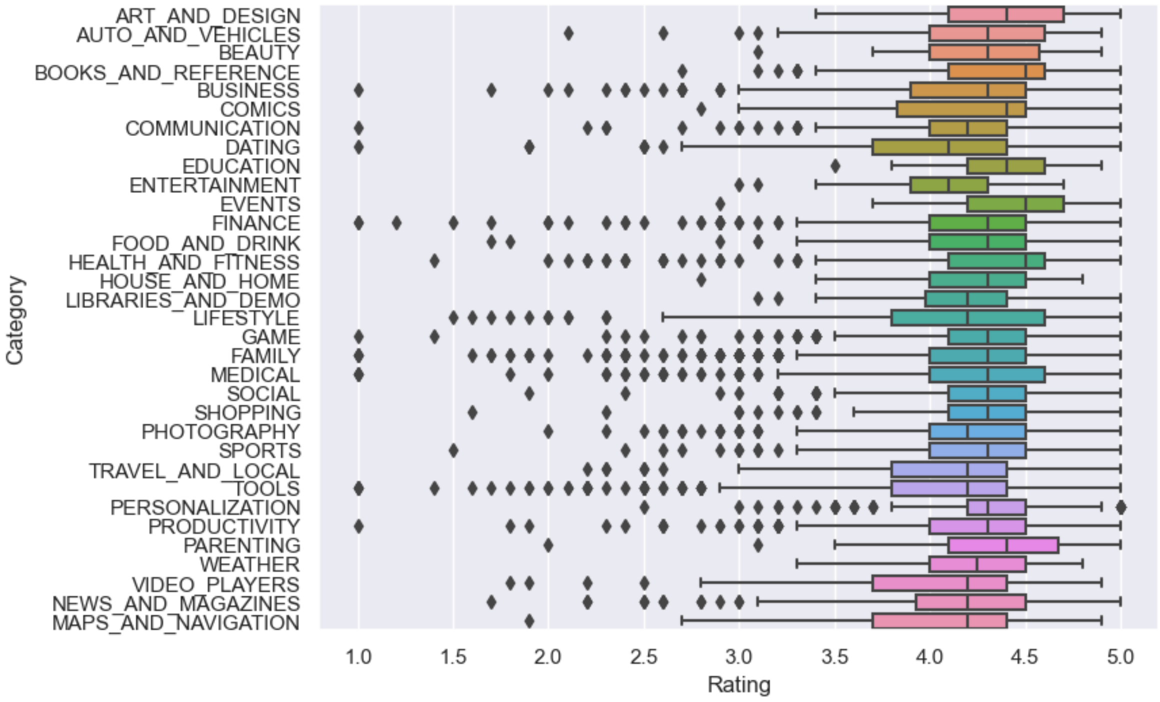
In [246]:

```
#5) Boxplot for Ratings Vs. Category

sns.boxplot(x="Rating", y="Category", data=data)
```

Out[246]:

```
<Axes: xlabel='Rating', ylabel='Category'>
```



From the above plot the Category Events has the best Ratings out of all other app genres

Data Preprocessing

# Model development

8(I).

In [247]:

```
#creating a copy of the data(df) to make all edits

inp1 = data
```

In [248]:

```
inp1.head()
```

Out[248]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up
5	Paper flowers instructions	ART_AND_DESIGN	4.4	167.0	5600.0	50000	Free	0	Everyone	Art & Design	March 26, 2017	1.0	2.3 and up

Reviews and Installs column still have some relatively high values,

before building the linear regression model we need to reduce the skew; columns needs log transformation"

In [249]:

```
inp1.skew()
```

```
C:\Users\Parag\AppData\Local\Temp\ipykernel_14540\3545313420.py:1: FutureWarning: The default value of numeric_only in DataFr
ame.skew is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is depreca
ted. Select only valid columns or specify the value of numeric_only to silence this warning.
  inp1.skew()
```

Out[249]:

```
Rating      -1.749753
Reviews      4.576494
Size         1.655917
Installs     1.543697
Price       18.074542
dtype: float64
```

In [250]:

```
##1) apply log transformation to Reviews
reviewskew = np.log1p(inp1['Reviews'])
inp1['Reviews'] = reviewskew
```

In [251]:

```
reviewskew.skew()
```

Out[251]:

```
-0.20039949659264134
```

In [252]:

```
##1 apply log transformation to Installs
installsskew = np.log1p(inp1['Installs'])
inp1['Installs']
```

Out[252]:

```
0           10000
1          500000
2         5000000
4          100000
5           50000
           ...
10834         500
10836        5000
10837         100
10839        1000
10840    10000000
Name: Installs, Length: 8496, dtype: int32
```

In [253]:

```
installsskew.skew()
```

Out[253]:

```
-0.5097286542754812
```

In [254]:

```
inp1.head()
```

Out[254]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	5.075174	19000.0	10000	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	6.875232	14000.0	500000	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	11.379520	8700.0	5000000	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	6.875232	2800.0	100000	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up
5	Paper flowers instructions	ART_AND_DESIGN	4.4	5.123964	5600.0	50000	Free	0	Everyone	Art & Design	March 26, 2017	1.0	2.3 and up

8(II)

In [255]:

```
#2) Dropping the columns- App, Last Updated, Current Ver, Type, & Andriod Ver as these won't be useful for our model

inp1.drop(['App','Last Updated','Current Ver','Android Ver','Type'], axis= 1, inplace = True)
```

In [256]:

```
inp1.head()
```

Out[256]:

	Category	Rating	Reviews	Size	Installs	Price	Content Rating	Genres
0	ART_AND_DESIGN	4.1	5.075174	19000.0	10000	0	Everyone	Art & Design
1	ART_AND_DESIGN	3.9	6.875232	14000.0	500000	0	Everyone	Art & Design;Pretend Play
2	ART_AND_DESIGN	4.7	11.379520	8700.0	5000000	0	Everyone	Art & Design
4	ART_AND_DESIGN	4.3	6.875232	2800.0	100000	0	Everyone	Art & Design;Creativity
5	ART_AND_DESIGN	4.4	5.123964	5600.0	50000	0	Everyone	Art & Design

In [257]:

```
inp1.shape
```

Out[257]:

```
(8496, 8)
```

As Model does not understand any Catergorical variable hence these need to be converted to numerical

Dummy Encoding is one way to convert these columns into numerical

8(III)

In [258]:

```
##3) create a copy of dataframe

inp2 = inp1
```

In [259]:

```
inp2.head()
```

Out[259]:

	Category	Rating	Reviews	Size	Installs	Price	Content Rating	Genres
0	ART_AND_DESIGN	4.1	5.075174	19000.0	10000	0	Everyone	Art & Design
1	ART_AND_DESIGN	3.9	6.875232	14000.0	500000	0	Everyone	Art & Design;Pretend Play
2	ART_AND_DESIGN	4.7	11.379520	8700.0	5000000	0	Everyone	Art & Design
4	ART_AND_DESIGN	4.3	6.875232	2800.0	100000	0	Everyone	Art & Design;Creativity
5	ART_AND_DESIGN	4.4	5.123964	5600.0	50000	0	Everyone	Art & Design

In [260]:

```
#get unique values in Column "Category"
inp2.Category.unique()
```

Out[260]:

```
array(['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'],
      dtype=object)
```

In [261]:

```
inp2.Category = pd.Categorical(inp2.Category)

x = inp2[['Category']]
del inp2['Category']

dummies = pd.get_dummies(x, prefix = 'Category')
inp2 = pd.concat([inp2,dummies], axis=1)
inp2.head()
```

Out[261]:

	Rating	Reviews	Size	Installs	Price	Content Rating	Genres	Category_ART_AND_DESIGN	Category_AUTO_AND_VEHICLES	Category_BEAUTY	..
0	4.1	5.075174	19000.0	10000	0	Everyone	Art & Design	1	0	0	..
1	3.9	6.875232	14000.0	500000	0	Everyone	Art & Design;Pretend Play	1	0	0	..
2	4.7	11.379520	8700.0	5000000	0	Everyone	Art & Design	1	0	0	..
4	4.3	6.875232	2800.0	100000	0	Everyone	Art & Design;Creativity	1	0	0	..
5	4.4	5.123964	5600.0	50000	0	Everyone	Art & Design	1	0	0	..

5 rows × 40 columns

In [262]:

```
inp2.shape
```

Out[262]:

```
(8496, 40)
```

In [263]:

```
#get unique values in Column "Genres"
inp2["Genres"].unique()
```

Out[263]:

```
array(['Art & Design', 'Art & Design;Pretend Play',
       'Art & Design;Creativity', 'Auto & Vehicles', 'Beauty',
       'Books & Reference', 'Business', 'Comics', 'Comics;Creativity',
       'Communication', 'Dating', 'Education', 'Education;Creativity',
       'Education;Education', 'Education;Music & Video',
       'Education;Action & Adventure', 'Education;Pretend Play',
       'Education;Brain Games', 'Entertainment',
       'Entertainment;Brain Games', 'Entertainment;Creativity',
       'Entertainment;Music & Video', 'Events', 'Finance', 'Food & Drink',
       'Health & Fitness', 'House & Home', 'Libraries & Demo',
       'Lifestyle', 'Lifestyle;Pretend Play', 'Card', 'Casual', 'Puzzle',
       'Action', 'Arcade', 'Word', 'Racing', 'Casual;Creativity',
       'Sports', 'Board', 'Simulation', 'Role Playing', 'Adventure',
       'Strategy', 'Simulation;Education', 'Action;Action & Adventure',
       'Trivia', 'Casual;Brain Games', 'Simulation;Action & Adventure',
       'Educational;Creativity', 'Puzzle;Brain Games',
       'Educational;Education', 'Card;Brain Games',
       'Educational;Brain Games', 'Educational;Pretend Play',
       'Casual;Action & Adventure', 'Entertainment;Education',
       'Casual;Education', 'Casual;Pretend Play', 'Music;Music & Video',
       'Racing;Action & Adventure', 'Arcade;Pretend Play',
       'Adventure;Action & Adventure', '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;Brain Games',
       'Parenting;Education', 'Weather', 'Video Players & Editors',
       'Video Players & Editors;Music & Video', 'News & Magazines',
       'Maps & Navigation', 'Health & Fitness;Action & Adventure',
       'Music', 'Educational', 'Casino', 'Adventure;Brain Games',
       'Lifestyle;Education', 'Books & Reference;Education',
       'Puzzle;Education', 'Role Playing;Brain Games',
       'Strategy;Education', 'Racing;Pretend Play',
       'Communication;Creativity', 'Strategy;Creativity'], dtype=object)
```

There are too many categories under Genres. Hence,

we will try to reduce some categories which have very few samples under them and put them under one new common category i.e. "Other

In [264]:

```
#Create an empty list

lists = []

#Get the total genres count and gernes count of perticular gerner count less than 20 append those into the list

for i in inp2.Genres.value_counts().index:
    if inp2.Genres.value_counts()[i]<20:
        lists.append(i)

#changing the gerners which are in the list to other

inp2.Genres = ['Other' if i in lists else i for i in inp2.Genres]
```

In [265]:

```
inp2["Genres"].unique()
```

Out[265]:

```
array(['Art & Design', 'Other', 'Auto & Vehicles', 'Beauty',
       'Books & Reference', 'Business', 'Comics', 'Communication',
       'Dating', 'Education', 'Education;Education',
       'Education;Pretend Play', 'Entertainment',
       'Entertainment;Music & Video', 'Events', 'Finance', 'Food & Drink',
       'Health & Fitness', 'House & Home', 'Libraries & Demo',
       'Lifestyle', 'Card', 'Casual', 'Puzzle', 'Action', 'Arcade',
       'Word', 'Racing', 'Sports', 'Board', 'Simulation', 'Role Playing',
       'Adventure', 'Strategy', 'Trivia', 'Educational;Education',
       'Casual;Pretend Play', 'Medical', 'Social', 'Shopping',
       'Photography', 'Travel & Local', 'Tools', 'Personalization',
       'Productivity', 'Parenting', 'Weather', 'Video Players & Editors',
       'News & Magazines', 'Maps & Navigation', 'Educational', 'Casino'],
      dtype=object)
```

In [266]:

```
#Storing the genres column into x varible and delete the genres col from dataframe inp2
#And concat the encoded cols to the dataframe inp2
inp2.Genres = pd.Categorical(inp2['Genres'])
x = inp2[["Genres"]]
del inp2['Genres']
dummies = pd.get_dummies(x, prefix = 'Genres')
inp2 = pd.concat([inp2,dummies], axis=1)
```

In [267]:

```
inp2.head()
```

Out[267]:

	Rating	Reviews	Size	Installs	Price	Content Rating	Category_ART_AND_DESIGN	Category_AUTO_AND_VEHICLES	Category_BEAUTY	Category_BOOKS_
0	4.1	5.075174	19000.0	10000	0	Everyone	1	0	0	
1	3.9	6.875232	14000.0	500000	0	Everyone	1	0	0	
2	4.7	11.379520	8700.0	5000000	0	Everyone	1	0	0	
4	4.3	6.875232	2800.0	100000	0	Everyone	1	0	0	
5	4.4	5.123964	5600.0	50000	0	Everyone	1	0	0	

5 rows × 91 columns

In [268]:

```
inp2.shape
```

Out[268]:

```
(8496, 91)
```

In [269]:

```
#get unique values in Column "Content Rating"
inp2["Content Rating"].unique()
```

Out[269]:

```
array(['Everyone', 'Teen', 'Everyone 10+', 'Mature 17+',
       'Adults only 18+', 'Unrated'], dtype=object)
```

In [270]:

```
#Applying one hot encoding
#Storing the Content Rating column into x varible and delete the Content Rating col from dataframe inp2
#And concat the encoded cols to the dataframe inp2

inp2['Content Rating'] = pd.Categorical(inp2['Content Rating'])

x = inp2[['Content Rating']]
del inp2['Content Rating']

dummies = pd.get_dummies(x, prefix = 'Content Rating')
inp2 = pd.concat([inp2,dummies], axis=1)
inp2.head()
```

Out[270]:

	Rating	Reviews	Size	Installs	Price	Category_ART_AND_DESIGN	Category_AUTO_AND_VEHICLES	Category_BEAUTY	Category_BOOKS_AND_REFE
0	4.1	5.075174	19000.0	10000	0	1	0	0	
1	3.9	6.875232	14000.0	500000	0	1	0	0	
2	4.7	11.379520	8700.0	5000000	0	1	0	0	
4	4.3	6.875232	2800.0	100000	0	1	0	0	
5	4.4	5.123964	5600.0	50000	0	1	0	0	

5 rows × 96 columns

In [271]:

```
inp2.shape
```

Out[271]:

```
(8496, 96)
```

9. and 10.

9.Train test split and apply 70-30 split. Name the new dataframes df_train and df_test.

10.Separate the dataframes into X_train, y_train, X_test, and y_test

In [279]:

```
#importing the neccessary libraries from sklearn to split the data and and for model building

from sklearn.model_selection import train_test_split as tts
from sklearn.linear_model import LinearRegression as LR
from sklearn.metrics import mean_squared_error as mse
```

In [280]:

```
#Creating the variable X and Y which contains the X features as independent features and Y is the target feature

d1 = inp2
X = d1.drop('Rating',axis=1)
y = d1['Rating']

#Dividing the X and y into test and train data

Xtrain, Xtest, ytrain, ytest = tts(X,y, test_size=0.3, random_state=5)
```

Model Building & Evaluation

11.

Model building Use linear regression as the technique Report the R2 on the train set

In [282]:

```
#Create a linear reggression obj by calling the linear reggressor algorithm

reg_all = LR()
reg_all.fit(Xtrain,ytrain)
```

Out[282]:

```
▾ LinearRegression
LinearRegression()
```

In [283]:

```
R2_train = round(reg_all.score(Xtrain,ytrain),3)
print("The R2 value of the Training Set is : {}".format(R2_train))
```

```
The R2 value of the Training Set is : 0.074
```

Make predictions on test set and report R2.

12.

Make predictions on test set and report R2.

In [286]:

```
# test the output by changing values, like 3750

R2_test = round(reg_all.score(Xtest,ytest),3)
print("The R2 value of the Testing Set is : {}".format(R2_test))
```

```
The R2 value of the Testing Set is : 0.063
```

In [ ]:

```

```