In [3]:
```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [5]:
```
data = pd.read_csv('googleplaystore1.csv')
```

In [6]:
```
data.head()
```

Out[6]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art &
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Design
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art &
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art &
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Design;(

In [7]:
```
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 [8]:
```
data.shape
```

Out[8]:
```
(10841, 13)
```

In [9]:
```
data.isnull().any()
```

Out[9]:
```
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 [10]:
```
data.isnull().sum()
```

Out[10]:
```
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
```

In [11]:
```
data = data.dropna()
```

In [12]:
```
data.isnull().any()
```

Out[12]:
```
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 [13]:
```
data.shape
```

Out[13]:
```
(9360, 13)
```

In [14]:
```
data["Size"] = [ float(i.split('M')[0]) if 'M' in i else float(0) for i in data["Siz
```

In [15]:
```
data.head()
```

Out[15]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0	Everyone	Art &
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0	Everyone	Design;
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0	Everyone	Art &
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0	Teen	Art &
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0	Everyone	Design;C

In [16]:
```
data["Size"] = 1000 * data["Size"]
```

In [17]:
```
data
```

Out[17]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Co R
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19000.0	10,000+	Free	0	Eve

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Co R
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14000.0	500,000+	Free	0	Eve
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8700.0	5,000,000+	Free	0	Eve
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25000.0	50,000,000+	Free	0	
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2800.0	100,000+	Free	0	Eve
...	...	...	...	...	...	...	...	...	
10834	FR Calculator	FAMILY	4.0	7	2600.0	500+	Free	0	Eve
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53000.0	5,000+	Free	0	Eve
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3600.0	100+	Free	0	Eve
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	0.0	1,000+	Free	0	M
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19000.0	10,000,000+	Free	0	Eve

9360 rows × 13 columns

In [18]:
```
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 [19]:
```
data["Reviews"] = data["Reviews"].astype(float)
```

In [20]:
```
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
```

In [21]:
```
data["Installs"] = [ float(i.replace('+','').replace(',', '')) if '+' in i or ',' in
```

In [22]:
```
data.head()
```

Out[22]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000.0	Free	0	Everyone	Ar
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000.0	Free	0	Everyone	Desi
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000.0	Free	0	Everyone	Ar

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000.0	Free	0	Teen	Ar
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000.0	Free	0	Everyone	Desig

In [23]:
```
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 [24]:
```
data["Installs"] = data["Installs"].astype(int)
```

In [25]:
```
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
```

In [26]:
```
data['Price'] = [ float(i.split('$')[1]) if '$' in i else float(0) for i in data['Pr
```

In [27]:
```
data.head()
```

Out[27]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000	Free	0.0	Everyone	Art
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000	Free	0.0	Everyone	Desig
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000	Free	0.0	Everyone	Art
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000	Free	0.0	Teen	Art
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000	Free	0.0	Everyone	Design;

In [28]:
```
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 [29]:
```
data["Price"] = data["Price"].astype(int)
```

In [30]:
```
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
```

In [31]:
```
data.shape
```

Out[31]:
```
(9360, 13)
```

In [32]:
```
data.drop(data[(data['Reviews'] < 1) & (data['Reviews'] > 5 )].index, inplace = True
```

In [33]:
```
data.shape
```

Out[33]:
```
(9360, 13)
```

In [34]:
```
data.shape
```

Out[34]:
```
(9360, 13)
```

In [35]:
```
data.drop(data[data['Installs'] < data['Reviews'] ].index, inplace = True)
```

In [36]:
```
data.shape
```

Out[36]:
```
(9353, 13)
```

In [37]:
```
data.shape
```

Out[37]:
```
(9353, 13)
```

In [38]:
```
data.drop(data[(data['Type'] =='Free') & (data['Price'] > 0 )].index, inplace = True
```

In [39]:
```
data.shape
```

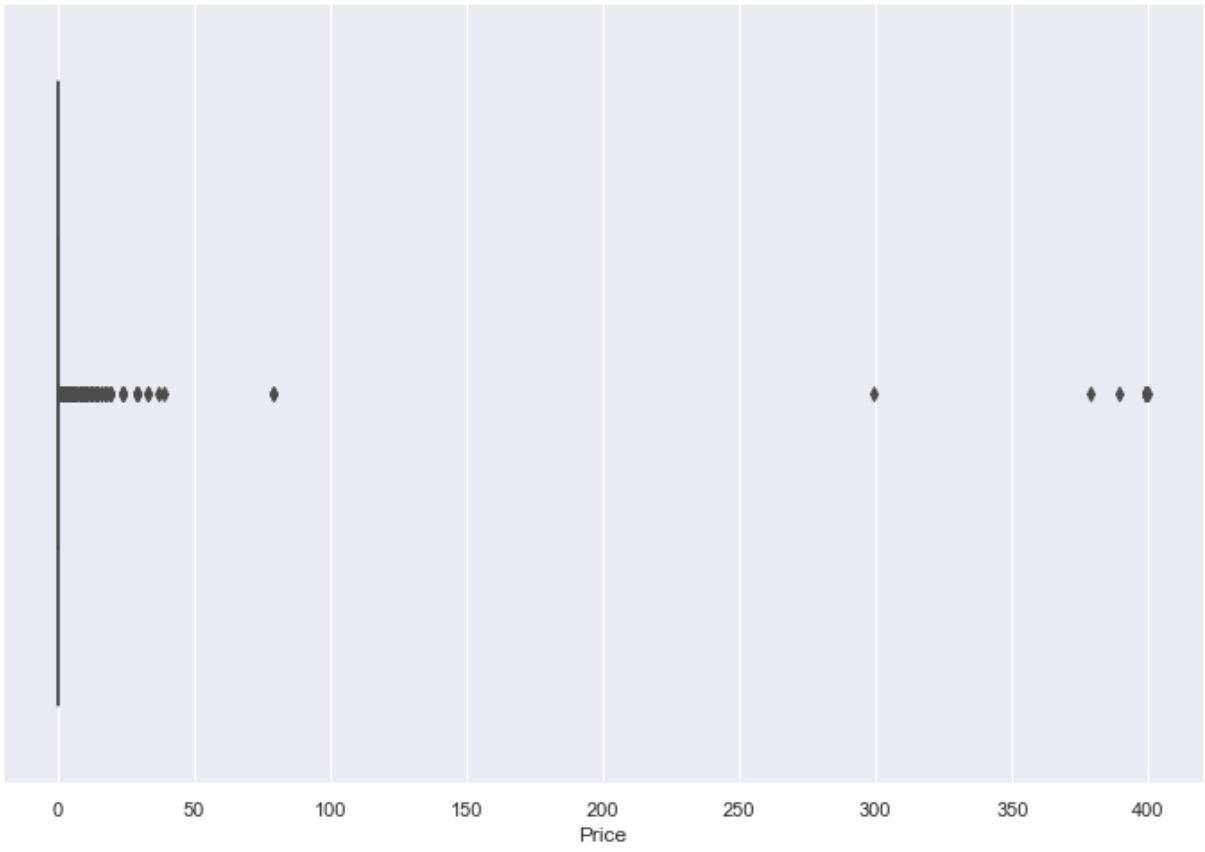
Out[39]:
```
(9353, 13)
```

In [40]:
```
sns.set(rc={'figure.figsize':(12,8)})
```

In [41]:
```
sns.boxplot(data['Price'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid p
ositional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

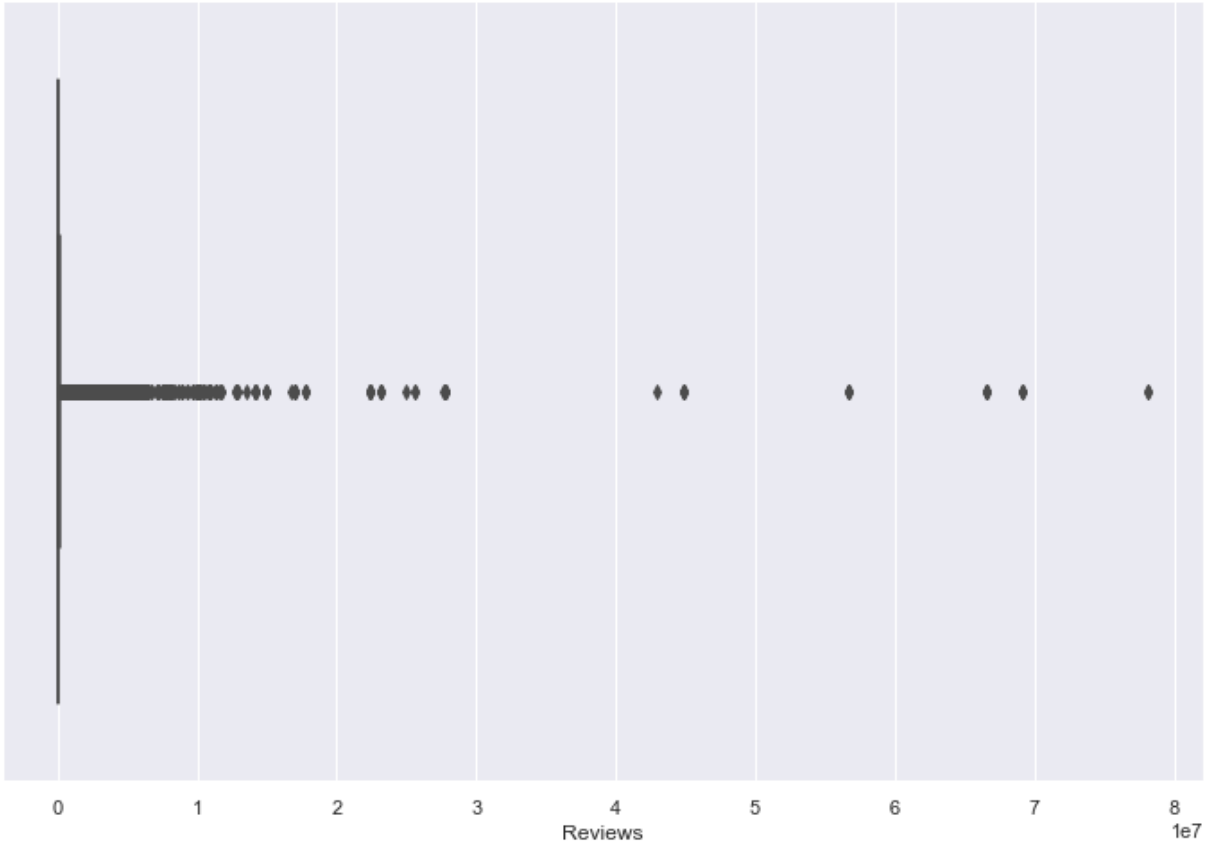
Out[41]:
```
<AxesSubplot:xlabel='Price'>
```



In [42]:
```
sns.boxplot(data['Reviews'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid p
ositional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

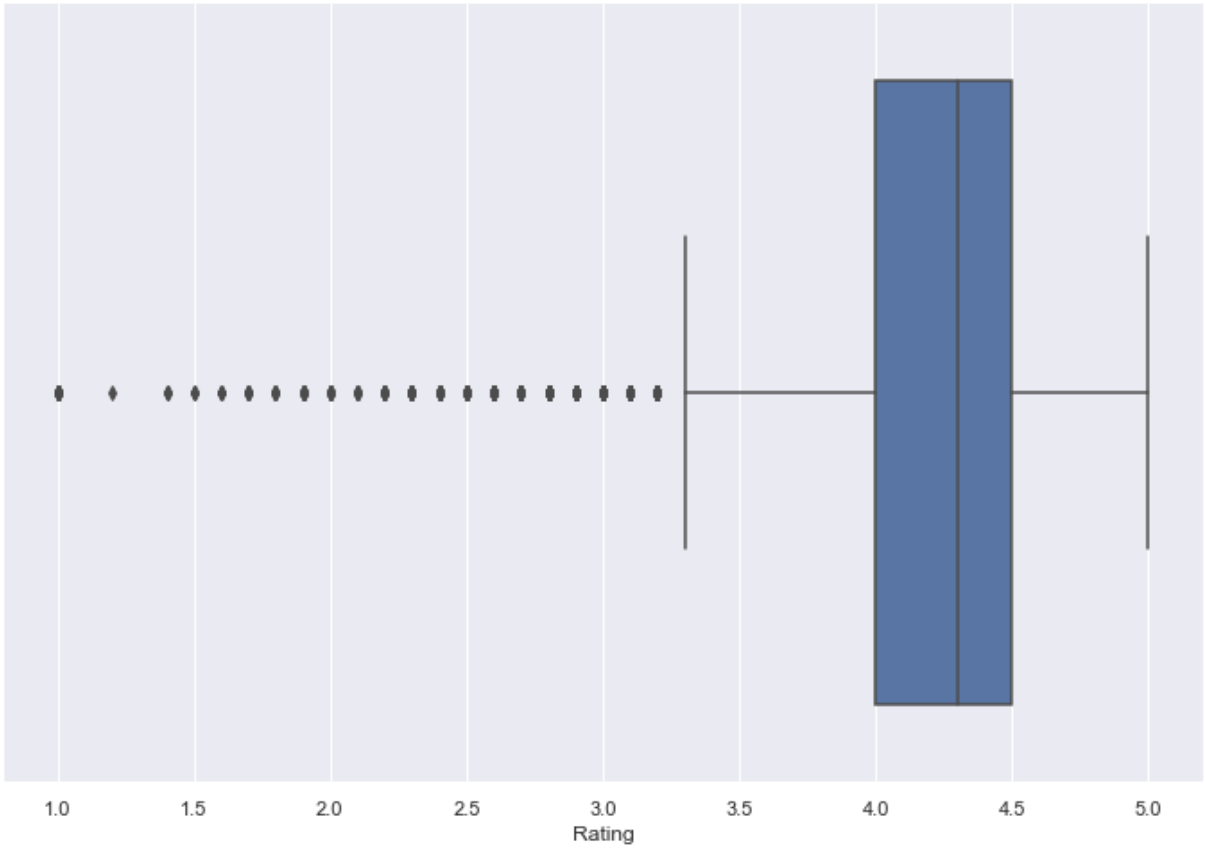
Out[42]:
```
<AxesSubplot:xlabel='Reviews'>
```



In [43]:
```
sns.boxplot(data['Rating'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid p
ositional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

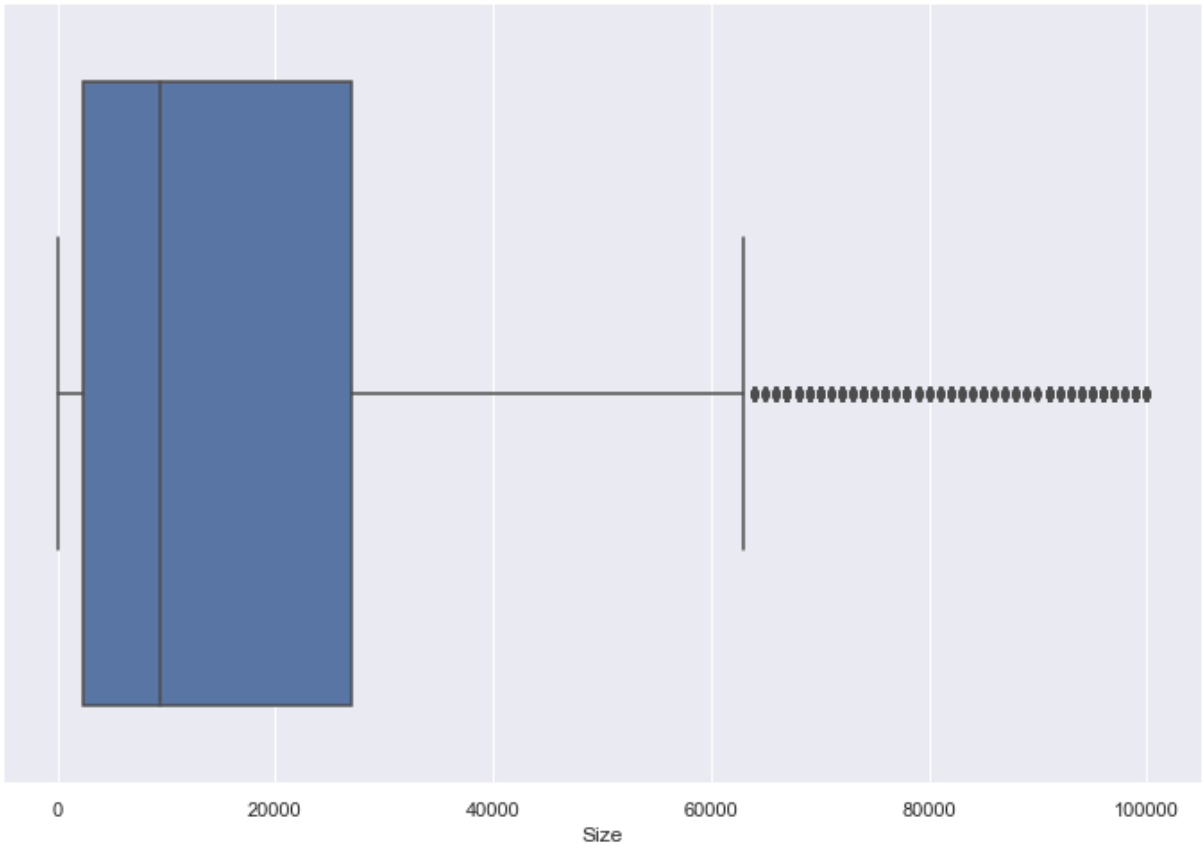
Out[43]:
```
<AxesSubplot:xlabel='Rating'>
```



In [44]:
```
sns.boxplot(data['Size'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid p
ositional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[44]:
```
<AxesSubplot:xlabel='Size'>
```



In [45]:
```
more = data.apply(lambda x : True
            if x['Price'] > 200 else False, axis = 1)
```

In [46]:
```
more_count = len(more[more == True].index)
```

In [47]:
```
data.shape
```

Out[47]:
```
(9353, 13)
```

In [48]:
```
data.drop(data[data['Price'] > 200].index, inplace = True)
```

In [49]:
```
data.shape
```

Out[49]:
```
(9338, 13)
```

In [50]:
```
data.drop(data[data['Reviews'] > 2000000].index, inplace = True)
```

In [51]:
```
data.shape
```

Out[51]:
```
(8885, 13)
```

In [52]:
```
data.quantile([.1, .25, .5, .70, .90, .95, .99], axis = 0)
```

Out[52]:

	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 [53]:
```
data.quantile([.1, .25, .5, .70, .90, .95, .99], axis = 0)
```

Out[53]:

	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

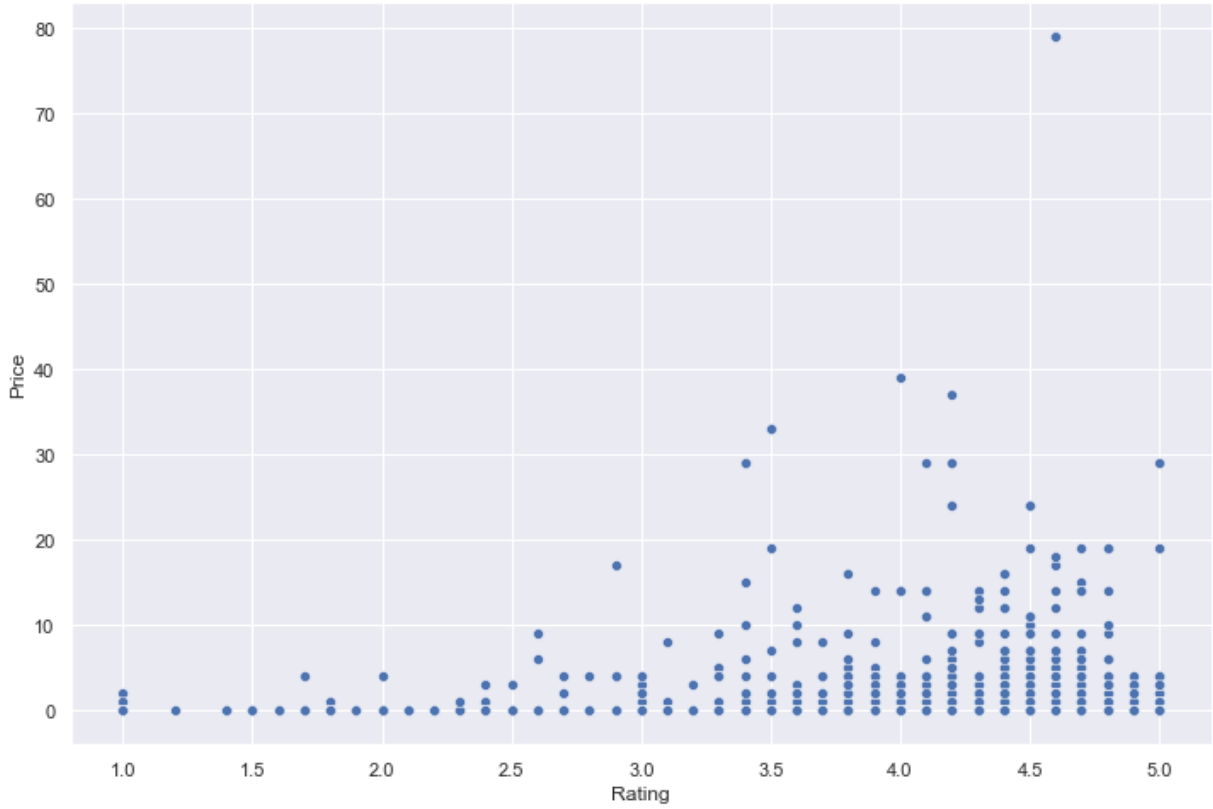
In [54]:
```
# dropping more than 10000000 Installs value
data.drop(data[data['Installs'] > 10000000].index, inplace = True)
```

In [55]:
```
data.shape
```

Out[55]:
```
(8496, 13)
```

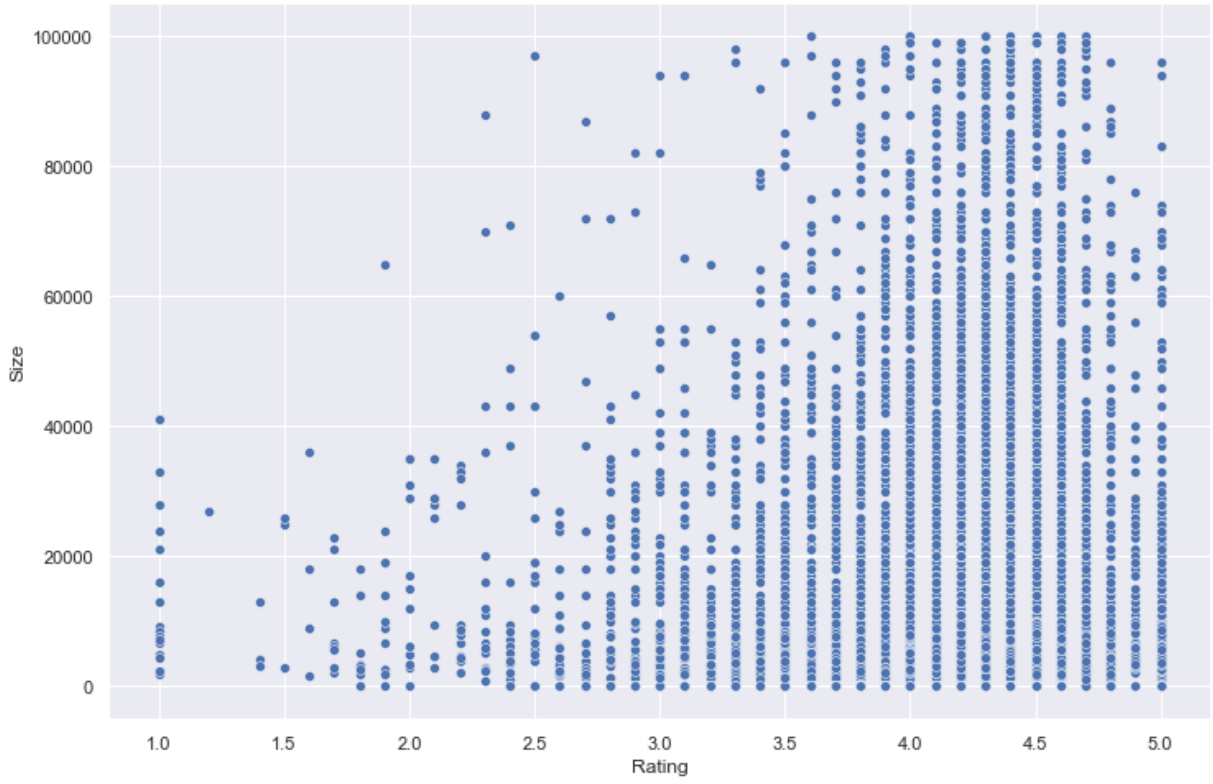
In [56]:
```
sns.scatterplot(x='Rating',y='Price',data=data)
```

Out[56]:
```
<AxesSubplot:xlabel='Rating', ylabel='Price'>
```



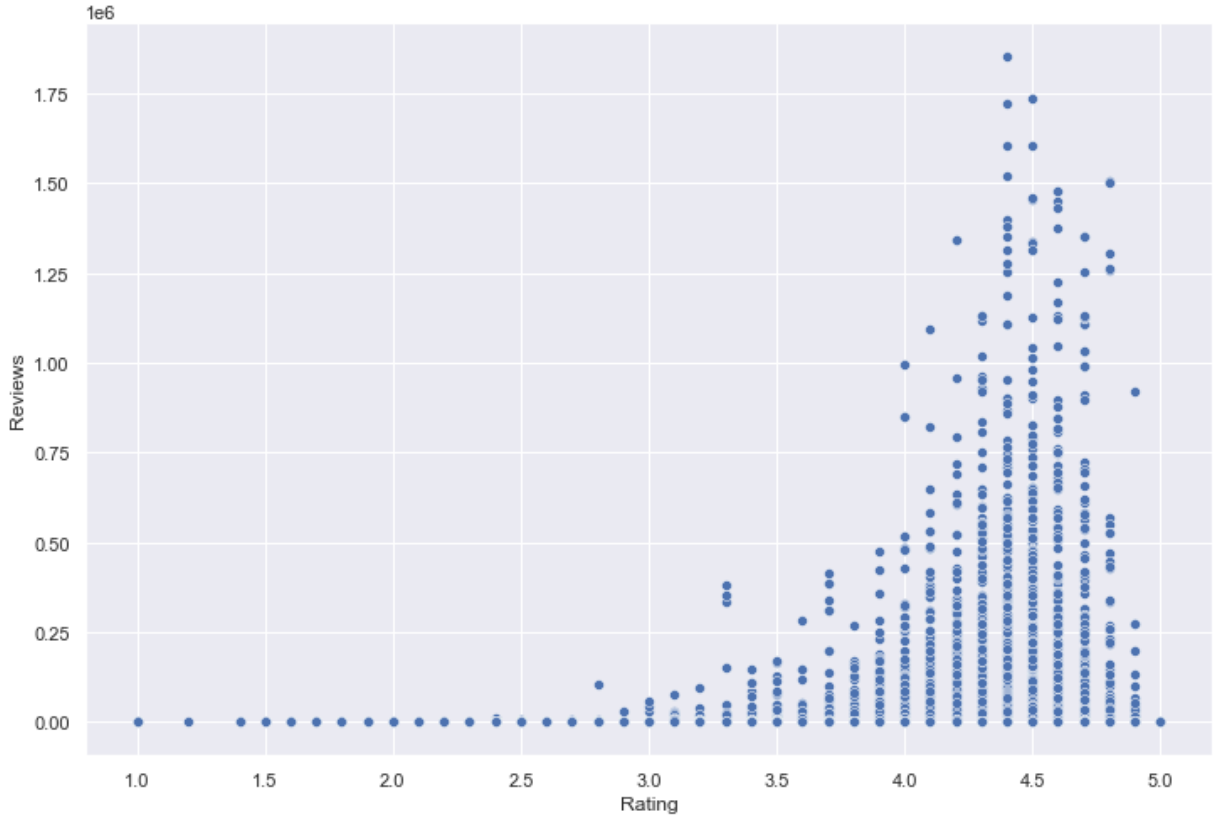
In [57]:
```
sns.scatterplot(x='Rating',y='Size',data=data)
```

Out[57]:
```
<AxesSubplot:xlabel='Rating', ylabel='Size'>
```



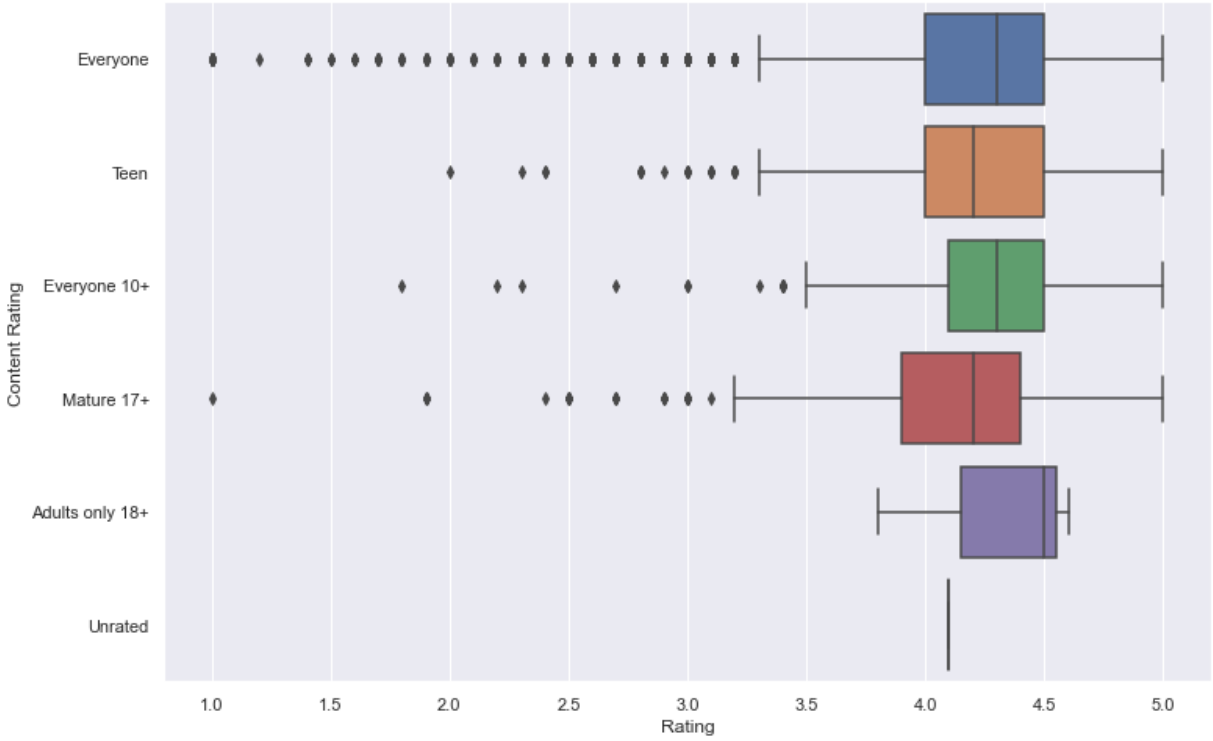
In [58]:
```
sns.scatterplot(x='Rating',y='Reviews',data=data)
```

Out[58]:
```
<AxesSubplot:xlabel='Rating', ylabel='Reviews'>
```



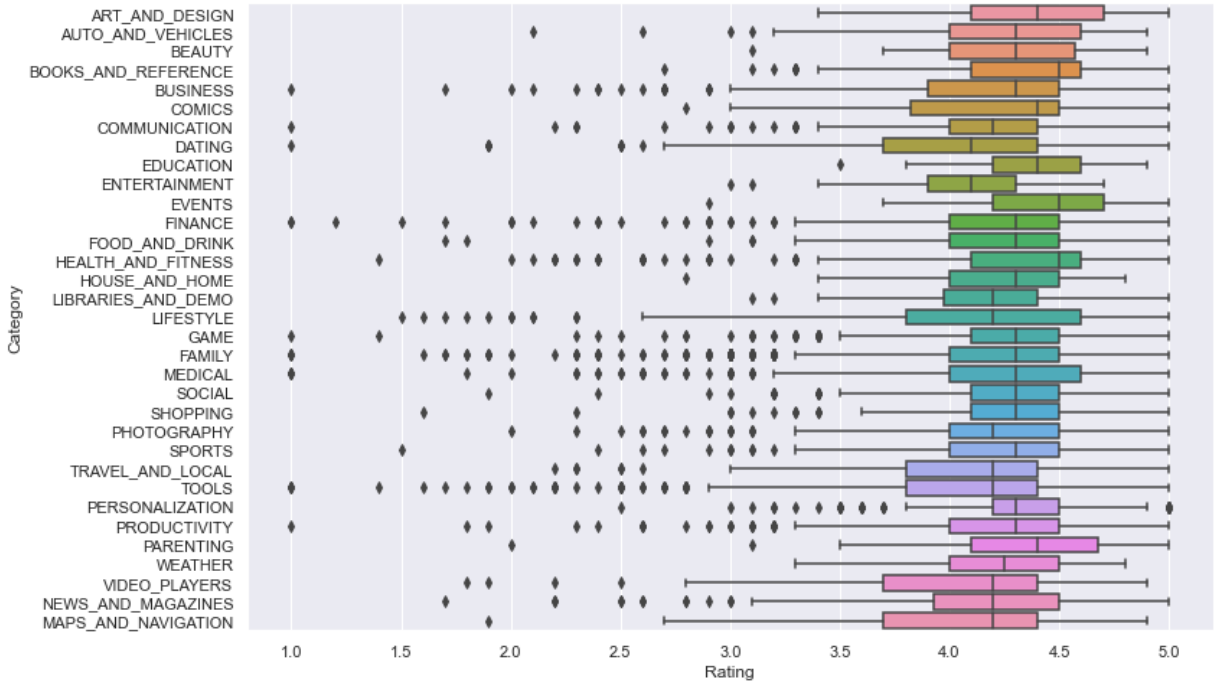
In [59]:
```
sns.boxplot(x="Rating", y="Content Rating", data=data)
```

Out[59]:
```
<AxesSubplot:xlabel='Rating', ylabel='Content Rating'>
```



In [60]:
```
sns.boxplot(x="Rating", y="Category", data=data)
```

Out[60]:
```
<AxesSubplot:xlabel='Rating', ylabel='Category'>
```



In [61]:
```
inp1 = data
```

In [62]:
```
inp1.head()
```

Out[62]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000	Free	0	Everyone	Art &
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000	Free	0	Everyone	Desigr
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8700.0	5000000	Free	0	Everyone	Art &
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2800.0	100000	Free	0	Everyone	Design;(
5	Paper flowers instructions	ART_AND_DESIGN	4.4	167.0	5600.0	50000	Free	0	Everyone	Art &

In [63]:
```
inp1.skew()
```

```
C:\Users\DELL\AppData\Local\Temp/ipykernel_10624/3545313420.py:1: FutureWarning: Dro
```

```
pping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is depr
ecated; in a future version this will raise TypeError.  Select only valid columns be
fore calling the reduction.
  inp1.skew()
```

Out[63]:
```
Rating      -1.749753
Reviews      4.576494
Size         1.655917
Installs     1.543697
Price       18.074542
dtype: float64
```

In [64]:
```
reviewskew = np.log1p(inp1['Reviews'])
inp1['Reviews'] = reviewskew
```

In [65]:
```
reviewskew.skew()
```

Out[65]:
```
-0.20039949659264134
```

In [66]:
```
installsskew = np.log1p(inp1['Installs'])
inp1['Installs']
```

Out[66]:
```
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 [67]:
```
installsskew.skew()
```

Out[67]:
```
-0.5097286542754812
```

In [68]:
```
inp1.head()
```

Out[68]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	5.075174	19000.0	10000	Free	0	Everyone	Ar
1	Coloring book moana	ART_AND_DESIGN	3.9	6.875232	14000.0	500000	Free	0	Everyone	Desi
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	11.379520	8700.0	5000000	Free	0	Everyone	Ar

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	6.875232	2800.0	100000	Free	0	Everyone	Desigr
5	Paper flowers instructions	ART_AND_DESIGN	4.4	5.123964	5600.0	50000	Free	0	Everyone	Ar

In [69]:
```
inp1.drop(["Last Updated","Current Ver","Android Ver","App","Type"],axis=1,inplace=T
```

In [70]:
```
inp1.head()
```

Out[70]:

	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 [71]:
```
inp1.shape
```

Out[71]:
```
(8496, 8)
```

In [72]:
```
inp2 = inp1
```

In [73]:
```
inp2.head()
```

Out[73]:

	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

Let's apply Dummy EnCoding on Column "Category"

In [74]:
```
#get unique values in Column "Category"
inp2.Category.unique()
```

Out[74]:
```
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 [75]:
```
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[75]:

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

5 rows × 40 columns

In [76]:
```
inp2.shape
```

Out[76]:
```
(8496, 40)
```

Let's apply Dummy EnCoding on Column "Genres"

In [77]:
```
#get unique values in Column "Genres"
inp2["Genres"].unique()
```

Out[77]:
```
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)
```

In [78]:

```
lists = []
for i in inp2.Genres.value_counts().index:
    if inp2.Genres.value_counts()[i]<20:
        lists.append(i)
inp2.Genres = ['Other' if i in lists else i for i in inp2.Genres]
```

In [79]:

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

Out[79]:

```
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 [80]:
```
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 [81]:
```
inp2.head()
```

Out[81]:

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

5 rows × 91 columns

In [82]:
```
inp2.shape
```

Out[82]:
```
(8496, 91)
```

Let's apply Dummy EnCoding on Column "Content Rating"

In [83]:
```
#get unique values in Column "Content Rating"
inp2["Content Rating"].unique()
```

Out[83]:
```
array(['Everyone', 'Teen', 'Everyone 10+', 'Mature 17+',
       'Adults only 18+', 'Unrated'], dtype=object)
```

In [84]:
```
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[84]:

	Rating	Reviews	Size	Installs	Price	Category_ART_AND_DESIGN	Category_AUTO_AND_VEH
0	4.1	5.075174	19000.0	10000	0	1	

	Rating	Reviews	Size	Installs	Price	Category_ART_AND_DESIGN	Category_AUTO_AND_VEI
1	3.9	6.875232	14000.0	500000	0	1	
2	4.7	11.379520	8700.0	5000000	0	1	
4	4.3	6.875232	2800.0	100000	0	1	
5	4.4	5.123964	5600.0	50000	0	1	

5 rows × 96 columns

In [85]:
```
inp2.shape
```

Out[85]:
```
(8496, 96)
```

In [86]:
```
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 [87]:
```
d1 = inp2
X = d1.drop('Rating',axis=1)
y = d1['Rating']

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

In [89]:
```
reg_all = LR()
reg_all.fit(Xtrain,ytrain)
```

Out[89]:
```
LinearRegression()
```

In [90]:
```
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
```

In [91]:
```
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
```

End Of The Project