In [222]:

```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings('ignore')
```

In [223]:

```
# Reading the Dataset
df=pd.read_csv(r'googleplaystores.csv')
```

In [224]:

```
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
```

In [225]:

```
# if any NA values are present, drop that row or column.
df.dropna()
```

Out[225]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0
...	...	...	...	...	...	...	...	...
10834	FR Calculator	FAMILY	4.0	7	2.6M	500+	Free	0
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	Varies with device	1,000+	Free	0
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19M	10,000,000+	Free	0

9360 rows × 13 columns

In [228]:

```
# replacing the empty positions
df.fillna(0,inplace=True)
```

In [229]:

```
# checks of null values
df.isnull().sum()
```

Out[229]:

```
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
```

In [ ]:

```
# Replacing K with *1000 <- Return it in float

# Replacing M with *1000000 <- Return it in float
def change(x):
    if type(x) == object or type(x) == int:
        return x
    if 'K' in x:
        if len(x) > 1:
            return float(x.replace('K', '')) * 1000
        return 1000.0
    if 'M' in x:
        if len(x) > 1:
            return float(x.replace('M', '')) * 1000000
        return 1000000.0

df['Size'] = df['Size'].apply(change)
```

1) What is the range of the size of most of the apps?

Ans: 100 MB

In [231]:

```
Smax=df['Size'].max()
print("Max range of apps is:",Smax)
```

```
Max range of apps is: 100000000.0
```

In [233]:

```
# Replacing the dollar '$' sign with space ' ' to avoid error
df['Price']=df['Price'].str.replace('$','')

# Converting price column values to Numeric
df['Price'] = pd.to_numeric(df['Price'],errors = 'coerce')
```

In [235]:

```
# Find the maximum Price Available
df['Price'].max()
```

Out[235]:

```
400.0
```

3)What is the app name having max price?

Ans : $400 I'm Rich - Trump Edition

In [234]:

```
CApp=df['App'].loc[df['Price']==400.0]
print(CApp)
```

```
4367    I'm Rich - Trump Edition
Name: App, dtype: object
```

In [236]:

```
# Replacing the plus '+' sign with space ' ' to avoid error
df['Installs']=df['Installs'].str.replace('+','')

# Replacing the comma ',' sign with space ' ' to avoid error
df['Installs'] = df['Installs'].str.replace(',','')
```

In [237]:

```
df = df.drop(10472)
# Converting Installs which is String to Numeric
df['Installs'] = pd.to_numeric(df['Installs'])
```

In [183]:

```
# Creating a separate dataframe for GAME Category

a=df.loc[df['Category'] =='GAME']
a
```

Out[183]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating
1653	ROBLOX	GAME	4.5	4447388	67000000.0	100000000	Free	0.0	Everyone 10+
1654	Subway Surfers	GAME	4.5	27722264	76000000.0	1000000000	Free	0.0	Everyone 10+
1655	Candy Crush Saga	GAME	4.4	22426677	74000000.0	500000000	Free	0.0	Everyone
1656	Solitaire	GAME	4.7	254258	23000000.0	10000000	Free	0.0	Everyone
1657	Bubble Shooter	GAME	4.5	148897	46000000.0	10000000	Free	0.0	Everyone
...	...	...	...	...	...	...	...	...	...
10791	Winter Wonderland	GAME	4.0	1287	38000000.0	50000	Free	0.0	Everyone
10792	Soccer Clubs Logo Quiz	GAME	4.2	21661	16000000.0	1000000	Free	0.0	Everyone
10793	Sid Story	GAME	4.4	28510	78000000.0	500000	Free	0.0	Teen
10803	Fatal Raid - No.1 Mobile FPS	GAME	4.3	56496	81000000.0	1000000	Free	0.0	Teen
10804	Poker Pro.Fr	GAME	4.2	5442	17000000.0	100000	Free	0.0	Teen

1144 rows × 13 columns

In [238]:

```
# Top 20 Apps with respect to no. of Installs in the Game Category ( including duplicates)

LApp=a.nlargest(20, ['Installs'])
print(LApp)
```

```
                 App Category  Rating   Reviews        Size    Installs  \
1654    Subway Surfers     GAME     4.5  27722264  76000000.0  1000000000
1700    Subway Surfers     GAME     4.5  27723193  76000000.0  1000000000
1750    Subway Surfers     GAME     4.5  27724094  76000000.0  1000000000
1872    Subway Surfers     GAME     4.5  27725352  76000000.0  1000000000
1917    Subway Surfers     GAME     4.5  27725352  76000000.0  1000000000
3896    Subway Surfers     GAME     4.5  27711703  76000000.0  1000000000
1655  Candy Crush Saga     GAME     4.4  22426677  74000000.0   500000000
1661      Temple Run 2     GAME     4.3   8118609  62000000.0   500000000
1662               Pou     GAME     4.3  10485308  24000000.0   500000000
1702               Pou     GAME     4.3  10485334  24000000.0   500000000
1705  Candy Crush Saga     GAME     4.4  22428456  74000000.0   500000000
1722    My Talking Tom     GAME     4.5  14891223         NaN   500000000
1729      Temple Run 2     GAME     4.3   8118937  62000000.0   500000000
1751  Candy Crush Saga     GAME     4.4  22428456  74000000.0   500000000
1759      Temple Run 2     GAME     4.3   8118937  62000000.0   500000000
1842  Candy Crush Saga     GAME     4.4  22429716  74000000.0   500000000
1869  Candy Crush Saga     GAME     4.4  22430188  74000000.0   500000000
1885               Pou     GAME     4.3  10486018  24000000.0   500000000
1886      Temple Run 2     GAME     4.3   8119151  62000000.0   500000000
1908    My Talking Tom     GAME     4.5  14892469         NaN   500000000

      Type  Price Content Rating  Genres   Last Updated Current Ver  \
1654  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
1700  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
1750  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
1872  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
1917  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
3896  Free    0.0   Everyone 10+  Arcade  July 12, 2018      1.90.0
1655  Free    0.0       Everyone  Casual   July 5, 2018   1.129.0.2
1661  Free    0.0       Everyone  Action   July 5, 2018      1.49.1
1662  Free    0.0       Everyone  Casual   May 25, 2018      1.4.77
1702  Free    0.0       Everyone  Casual   May 25, 2018      1.4.77
1705  Free    0.0       Everyone  Casual   July 5, 2018   1.129.0.2
1722  Free    0.0       Everyone  Casual  July 19, 2018   4.8.0.132
1729  Free    0.0       Everyone  Action   July 5, 2018      1.49.1
1751  Free    0.0       Everyone  Casual   July 5, 2018   1.129.0.2
1759  Free    0.0       Everyone  Action   July 5, 2018      1.49.1
1842  Free    0.0       Everyone  Casual   July 5, 2018   1.129.0.2
1869  Free    0.0       Everyone  Casual   July 5, 2018   1.129.0.2
1885  Free    0.0       Everyone  Casual   May 25, 2018      1.4.77
1886  Free    0.0       Everyone  Action   July 5, 2018      1.49.1
1908  Free    0.0       Everyone  Casual  July 19, 2018   4.8.0.132

     Android Ver
1654  4.1 and up
1700  4.1 and up
1750  4.1 and up
1872  4.1 and up
1917  4.1 and up
3896  4.1 and up
1655  4.1 and up
1661  4.0 and up
```

```
1662  4.0 and up
1702  4.0 and up
1705  4.1 and up
1722  4.1 and up
1729  4.0 and up
1751  4.1 and up
1759  4.0 and up
1842  4.1 and up
1869  4.1 and up
1885  4.0 and up
1886  4.0 and up
1908  4.1 and up
```

In [239]:

```
#List of APPS in the game category
x1=a['App'].loc[a['Category']=='GAME']
x1
```

Out[239]:

```
1653                          ROBLOX
1654                  Subway Surfers
1655                 Candy Crush Saga
1656                        Solitaire
1657                  Bubble Shooter
                     ...
10791               Winter Wonderland
10792          Soccer Clubs Logo Quiz
10793                       Sid Story
10803    Fatal Raid - No.1 Mobile FPS
10804                    Poker Pro.Fr
Name: App, Length: 1144, dtype: object
```

In [289]:

```
# Apps in the Game Category which have a minimum of 500000000 downloads

x1=a['App'].loc[a['Installs']>=500000000]
print("Top 5 Apps based on Installation\n:",x1.unique())
```

```
Top 5 Apps based on Installation
: ['Subway Surfers' 'Candy Crush Saga' 'Temple Run 2' 'Pou' 'My Talking To
m']
```

In [290]:

```
# Apps in the Game Category which have a minimum of 100000000 downloads

x2=a['App'].loc[a['Installs']==100000000]
print(x2.unique())
```

```
['ROBLOX' 'slither.io' 'Clash Royale' 'Helix Jump' 'Angry Birds Rio'
 'Plants vs. Zombies FREE' 'Sonic Dash' 'Candy Crush Soda Saga'
 'Clash of Clans' 'PAC-MAN' '8 Ball Pool' 'Angry Birds Classic'
 'Flow Free' 'Zombie Tsunami' 'Hill Climb Racing'
 'Minion Rush: Despicable Me Official Game' 'Farm Heroes Saga'
 'My Talking Angela' 'Cut the Rope FULL FREE'
 'Sniper 3D Gun Shooter: Free Shooting Games - FPS' 'Cooking Fever'
 'Score! Hero' 'Garena Free Fire' 'Roll the Ball® - slide puzzle'
 'Talking Tom Gold Run' 'Dream League Soccer 2018' 'Traffic Racer'
 'Hill Climb Racing 2' 'Hungry Shark Evolution' 'Piano Tiles 2™'
 'Pokémon GO' 'Extreme Car Driving Simulator' 'Trivia Crack'
 'Angry Birds 2' 'Yes day' 'Crossy Road' 'Shadow Fight 2' 'Agar.io'
 'Bus Rush: Subway Edition' 'Jetpack Joyride' 'Super Mario Run'
 'Glow Hockey' 'Asphalt 8: Airborne' "Lep's World 2 ❁ ❁" 'Fruit Ninja®'
 'Vector' 'Dr. Driving' 'Bike Race Free - Top Motorcycle Racing Games'
 'Smash Hit' 'Temple Run' 'Geometry Dash Lite'
 'Ant Smasher by Best Cool & Fun Games' 'Angry Birds Star Wars'
 'Mobile Legends: Bang Bang' 'Banana Kong' 'Skater Boy'
 'Modern Combat 5: eSports FPS']
```

In [331]:

```
# Selecting a 15 random apps

random_selected = x2.sample(n=15)
print(random_selected.unique())
```

```
['Helix Jump' 'Shadow Fight 2' 'Clash of Clans'
 'Mobile Legends: Bang Bang' 'Hill Climb Racing' 'Banana Kong'
 'Plants vs. Zombies FREE' 'Garena Free Fire' 'Angry Birds Star Wars'
 'Angry Birds Classic' 'My Talking Angela'
 'Minion Rush: Despicable Me Official Game' 'Hungry Shark Evolution'
 'Cut the Rope FULL FREE']
```

In [346]:

```
# Storing the Unique values in variables

a1=(x1.unique())
a2=(random_selected.unique())
```

4)List top 20 apps in Game Category(Based on Number of Installation)

In [348]:

```
# Combining the variables using np.concatenate

frames = [a1,a2]
result = np.concatenate(frames)
display(result)
```

```
array(['Subway Surfers', 'Candy Crush Saga', 'Temple Run 2', 'Pou',
       'My Talking Tom', 'Helix Jump', 'Shadow Fight 2', 'Clash of Clans',
       'Mobile Legends: Bang Bang', 'Hill Climb Racing', 'Banana Kong',
       'Plants vs. Zombies FREE', 'Garena Free Fire',
       'Angry Birds Star Wars', 'Angry Birds Classic',
       'My Talking Angela', 'Minion Rush: Despicable Me Official Game',
       'Hungry Shark Evolution', 'Cut the Rope FULL FREE'], dtype=object)
```

In [ ]:

```

```