In [2]:
```
import csv
dataset = list(csv.reader(open("AppleStore.csv", encoding = "utf-8")))
```

In [3]:
```
dataset
```

Out[3]:
```
[['id',
  'track_name',
  'size_bytes',
  'currency',
  'price',
  'rating_count_tot',
  'rating_count_ver',
  'user_rating',
  'user_rating_ver',
  'ver',
  'cont_rating',
  'prime_genre',
  'sup_devices.num',
  'ipadSc_urls.num',
  'lang.num',
  'vpp_lic'],
 ['284882215',
  'Facebook',
  '389879808',
```

In [4]:
```
type(dataset)
```

Out[4]:
```
list
```

In [5]:
```
len(dataset)
```

Out[5]:
```
7198
```

In [5]:
```
dataset[0]
```

Out[5]:
```
['id',
 'track_name',
 'size_bytes',
 'currency',
 'price',
 'rating_count_tot',
 'rating_count_ver',
 'user_rating',
 'user_rating_ver',
 'ver',
 'cont_rating',
 'prime_genre',
 'sup_devices.num',
 'ipadSc_urls.num',
 'lang.num',
 'vpp_lic']
```

In [6]:
```
dataset[1]
```

Out[6]:
```
['284882215',
 'Facebook',
 '389879808',
 'USD',
 '0',
 '2974676',
 '212',
 '3.5',
 '3.5',
 '95',
 '4+',
 'Social Networking',
 '37',
 '1',
 '29',
 '1']
```

In [6]:
```
header=dataset[0]
header
```

Out[6]:
```
['id',
 'track_name',
 'size_bytes',
 'currency',
 'price',
 'rating_count_tot',
 'rating_count_ver',
 'user_rating',
 'user_rating_ver',
 'ver',
 'cont_rating',
 'prime_genre',
 'sup_devices.num',
 'ipadSc_urls.num',
 'lang.num',
 'vpp_lic']
```

In [7]:
```
data=dataset[1:]
data
```

Out[7]:
```
[['284882215',
  'Facebook',
  '389879808',
  'USD',
  '0',
  '2974676',
  '212',
  '3.5',
  '3.5',
  '95',
  '4+',
  'Social Networking',
  '37',
  '1',
  '29',
  '1'],
 ['389801252',
  'Instagram',
  '113954816',
```

In [8]:
```
#fetching specific column
for i in dataset:
    print(i[1])
```

```
track_name
Facebook
Instagram
Clash of Clans
Temple Run
Pandora - Music & Radio
Pinterest
Bible
Candy Crush Saga
Spotify Music
Angry Birds
Subway Surfers
Fruit Ninja Classic
Solitaire
CSR Racing
Crossy Road - Endless Arcade Hopper
Injustice: Gods Among Us
Hay Day
Clear Vision (17+)
```

In [15]:
```
len(dataset)
```

Out[15]:
```
7198
```

In [12]:
```
# free and paid apps avg rating
free_apps_rating = []
paid_apps_rating = []
for i in dataset[1:]:
    price = float(i[4]) # 0.99
    rating = float(i[7]) # 3.5
    if price == 0: # 0.99 == 0: False
        free_apps_rating.append(rating)
#print("Free App: ", price, rating)
    else:
        paid_apps_rating.append(rating)
#print("Paid App: ", price, rating)
```

In [13]:
```
free_apps_rating
```

...

In [14]:
```
print(round(sum(free_apps_rating)/len(free_apps_rating),2))
print(sum(paid_apps_rating)/len(paid_apps_rating))
```

```
3.38
3.720948742438714
```

In [23]:
```
free_apps=[]
paid_apps=[]
for i in data:
        price = float(i[4])
        if price == 0:
            free_apps.append(i[1])
        else:
            paid_apps.append(i[1])
```

In [24]:
```
free_apps
```

Out[24]:
```
['Facebook',
 'Instagram',
 'Clash of Clans',
 'Temple Run',
 'Pandora - Music & Radio',
 'Pinterest',
 'Bible',
 'Candy Crush Saga',
 'Spotify Music',
 'Angry Birds',
 'Subway Surfers',
 'Solitaire',
 'CSR Racing',
 'Crossy Road - Endless Arcade Hopper',
 'Injustice: Gods Among Us',
 'Hay Day',
 'PAC-MAN',
 'Calorie Counter & Diet Tracker by MyFitnessPal',
 'DragonVale',
```

In [15]:
```
#finding each categories count of apps
cat_name_counts = {}

# Iterate through the data and count app names
for i in data:
    cat_name = i[11]  # App name is at index 1 in each sublist
    if cat_name in cat_name_counts:
        cat_name_counts[cat_name] += 1
    else:
        cat_name_counts[cat_name] = 1
```

In [17]:
```
cat_name_counts.keys()
```

Out[17]:
```
dict_keys(['Social Networking', 'Photo & Video', 'Games', 'Music', 'Refere
nce', 'Health & Fitness', 'Weather', 'Utilities', 'Travel', 'Shopping', 'N
ews', 'Navigation', 'Lifestyle', 'Entertainment', 'Food & Drink', 'Sport
s', 'Book', 'Finance', 'Education', 'Productivity', 'Business', 'Catalog
s', 'Medical'])
```

In [35]:
```
dict(sorted(cat_name_counts.items(),key = lambda x: x[1]))
```

Out[35]:
```
{'Catalogs': 10,
 'Medical': 23,
 'Navigation': 46,
 'Business': 57,
 'Food & Drink': 63,
 'Reference': 64,
 'Weather': 72,
 'News': 75,
 'Travel': 81,
 'Finance': 104,
 'Book': 112,
 'Sports': 114,
 'Shopping': 122,
 'Music': 138,
 'Lifestyle': 144,
 'Social Networking': 167,
 'Productivity': 178,
 'Health & Fitness': 180,
 'Utilities': 248,
 'Photo & Video': 349,
 'Education': 453,
 'Entertainment': 535,
 'Games': 3862}
```

In [18]:
```
#finding each categorie's avg rating
cat_name_counts = {}

# Iterate through the data and count app names
for i in data:
    cat_name = i[11]  # App name is at index 1 in each sublist
    if cat_name in cat_name_counts:
        cat_name_counts[cat_name] += 1
    else:
        cat_name_counts[cat_name] = 1
```

In [39]:
```
ratings=[]
for i in data:
    rating=i[7]
    category=i[11]
    if category=="Social Networking":
        ratings.append(float(i[7]))
```

In [40]:
```
sum(ratings)/len(ratings)
```

Out[40]:
```
2.9850299401197606
```

In [12]:
```
#find out all categories of cont_rating
header.index("cont_rating")
cont_rating_count={}
count=0
for i in data:
    cont_rating=i[10]
    if cont_rating not in cont_rating_count:
        cont_rating_count[cont_rating]=1
    else:
        cont_rating_count[cont_rating]+=1

cont_rating_count
```

Out[12]:
```
{'4+': 4433, '12+': 1155, '9+': 987, '17+': 622}
```

In [17]:
```
ages=list(cont_rating_count.keys())
ages
```

Out[17]:
```
['4+', '12+', '9+', '17+']
```

TASKS

1. Top three games that are popular among teens.
2. Report five most downloaded applications in social media category.
3. How many applications have never been rated.
4. Report top paid application of each category.
5. Which are the top three categories famous among adults.
6. Report all applications of "google"

In [ ]: