1. Introduction

Mobile apps are everywhere. They are easy to create and can be very lucrative from the business standpoint. Specifically, Android is expanding as an operating system and has captured more than 74% of the total market[1] (https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009).

The Google Play Store apps data has enormous potential to facilitate data-driven decisions and insights for businesses. In this notebook, we will analyze the Android app market by comparing ~10k apps in Google Play across different categories. We will also use the user reviews to draw a qualitative comparision between the apps.

The dataset you will use here was scraped from Google Play Store in September 2018 and was published on Kaggle (https://www.kaggle.com/lava18/google-play-store-apps). Here are the details:

datasets/apps.csv

This file contains all the details of the apps on Google Play. There are 9 features that describe a given app.

- **App:** Name of the app
- **Category:** Category of the app. Some examples are: ART_AND_DESIGN, FINANCE, COMICS, BEAUTY etc.
- **Rating:** The current average rating (out of 5) of the app on Google Play
- **Reviews:** Number of user reviews given on the app
- **Size:** Size of the app in MB (megabytes)
- **Installs:** Number of times the app was downloaded from Google Play
- **Type:** Whether the app is paid or free
- **Price:** Price of the app in US$
- **Last Updated:** Date on which the app was last updated on Google Play

datasets/user_reviews.csv

This file contains a random sample of 100 *[most helpful first]* *(https://www.androidpolice.com/2019/01/21/google-play-stores-redesigned-ratings-and-reviews-*

section-lets-you-easily-filter-by-star-rating/) user reviews for each app. The text in each review has been pre-processed and passed through a sentiment analyzer.

- **App:** Name of the app on which the user review was provided. Matches the \`App\` column of the \`apps.csv\` file
- **Review:** The pre-processed user review text
- **Sentiment Category:** Sentiment category of the user review - Positive, Negative or Neutral
- **Sentiment Score:** Sentiment score of the user review. It lies between [-1,1]. A higher score denotes a more positive sentiment.

From here on, it will be your task to explore and manipulate the data until you are able to answer the three questions described in the instructions panel.

In [15]:

```
#intialize pandas and numpy
import numpy as np
import pandas as pd
```

In [16]:

```
#Task 1: Read the apps.csv file and clean the Installs column to convert
#it into integer data type. Save your answer as a DataFrame apps.
#Going forward, you will do all your analysis on the apps DataFrame..

#load csv
apps = pd.read_csv("datasets/apps.csv")

#view dataframe
print(apps.head())
```

```
                                                 App        Category  R
ating  \
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     Last Updated
0      159  19.0      10,000+  Free    0.0  January 7, 2018
1      967  14.0     500,000+  Free    0.0 January 15, 2018
2    87510   8.7   5,000,000+  Free    0.0   August 1, 2018
3   215644  25.0  50,000,000+  Free    0.0     June 8, 2018
4      967   2.8     100,000+  Free    0.0    June 20, 2018
```

In [17]:

```
#need to get info on "Installs" column and determine what data type it
#is now
print(apps.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9659 entries, 0 to 9658
Data columns (total 9 columns):
App               9659 non-null object
Category          9659 non-null object
Rating            8196 non-null float64
Reviews           9659 non-null int64
Size              8432 non-null float64
Installs          9659 non-null object
Type              9659 non-null object
Price             9659 non-null float64
Last Updated      9659 non-null object
dtypes: float64(3), int64(1), object(5)
memory usage: 679.2+ KB
None
```

In [18]:

```
#need to remove symbols in "Installs" before converting into int

#replace '+' symbol with ''
apps["Installs"] = apps["Installs"].str.replace('+','')

#replace ',' symbol with ''
apps["Installs"] = apps["Installs"].str.replace(',','')

#view "Installs"
print(apps["Installs"].head())
```

```
0       10000
1      500000
2     5000000
3    50000000
4      100000
Name: Installs, dtype: object
```

In [19]:

```
#the '+' symbol has been removed. Convert "Installs" from object to int
apps["Installs"] = apps["Installs"].astype(int)

#verify results
print(apps["Installs"].head())
```

```
0       10000
1      500000
2     5000000
3    50000000
4      100000
Name: Installs, dtype: int64
```

In [20]:

```
#Task 2: Find the number of apps in each category, the average price,
#and the average rating. Save your answer as a DataFrame
#app_category_info. Your should rename the four columns
#as: Category, Number of apps, Average price, Average rating.

#Define a new data frame to store info ie. "Category", "Number of apps"
#"Average Price", "Average Rating"
#Use existing apps df, group by category and apply different summary
#statistic functions to columns using .agg()
app_category_info = apps.groupby("Category").agg({'App': 'count',
                                                 'Price': 'mean',
                                                 'Rating': 'mean'})

#verify results
#print(app_category_info.head())
```

In [21]:

```
#Change the existing column names to the required column names
#Use inplace = True to apply change onto current dataframe
app_category_info.rename(columns = {'App': 'Number of apps',
                                     'Price': 'Average price',
                                    'Rating': 'Average rating'},
                                     inplace = True)

#Remove index so 'Category' becomes a seperate column as per requirement
s
app_category_info.reset_index(level= 'Category', inplace = True)

#view results
print(app_category_info[:])
```

```
               Category  Number of apps  Average price  Average rating
0        ART_AND_DESIGN              64       0.093281        4.357377
1     AUTO_AND_VEHICLES              85       0.158471        4.190411
2                BEAUTY              53       0.000000        4.278571
3    BOOKS_AND_REFERENCE            222       0.539505        4.344970
4              BUSINESS             420       0.417357        4.098479
5                COMICS              56       0.000000        4.181481
6         COMMUNICATION             315       0.263937        4.121484
7                DATING             171       0.160468        3.970149
8             EDUCATION             119       0.150924        4.364407
9         ENTERTAINMENT             102       0.078235        4.135294
10               EVENTS              64       1.718594        4.435556
11               FAMILY            1832       1.309967        4.179664
12              FINANCE             345       8.408203        4.115563
13        FOOD_AND_DRINK            112       0.075714        4.172340
14                 GAME             959       0.296465        4.247368
15    HEALTH_AND_FITNESS            288       0.223438        4.243033
16        HOUSE_AND_HOME             74       0.000000        4.150000
17    LIBRARIES_AND_DEMO             84       0.011786        4.178125
18            LIFESTYLE             369       6.398022        4.093355
19   MAPS_AND_NAVIGATION            131       0.205725        4.036441
20              MEDICAL             395       2.520759        4.166552
21    NEWS_AND_MAGAZINES            254       0.015669        4.121569
22            PARENTING              60       0.159667        4.300000
23      PERSONALIZATION             376       0.400213        4.332215
24          PHOTOGRAPHY             281       0.420925        4.157414
25         PRODUCTIVITY             374       0.670936        4.183389
26             SHOPPING             202       0.027129        4.230000
27               SOCIAL             239       0.066820        4.247291
28               SPORTS             325       0.307692        4.216154
29                TOOLS             827       0.323156        4.039554
30      TRAVEL_AND_LOCAL            219       0.228082        4.069519
31        VIDEO_PLAYERS             163       0.064172        4.044595
32              WEATHER              79       0.410380        4.243056
```

In [22]:

```
#Task 3: Find the top 10 free FINANCE apps having the highest average
#sentiment score. Save your answer as a DataFrame top_10_user_feedback.
#Your answer should have exactly 10 rows and two columns named:
#App and Sentiment Score, where the average Sentiment Score is sorted
#from highest to lowest.

#load user_reviews.csv
user_reviews = pd.read_csv("datasets/user_reviews.csv").dropna()

#verify results and no missing data
user_reviews
```

Out[22]:

	App	Review	Sentiment Category	Sentiment Score
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.000000
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.250000
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.400000
4	10 Best Foods for You	Best idea us	Positive	1.000000
5	10 Best Foods for You	Best way	Positive	1.000000
6	10 Best Foods for You	Amazing	Positive	0.600000
8	10 Best Foods for You	Looking forward app,	Neutral	0.000000
9	10 Best Foods for You	It helpful site ! It help foods get !	Neutral	0.000000
10	10 Best Foods for You	good you.	Positive	0.700000
11	10 Best Foods for You	Useful information The amount spelling errors ...	Positive	0.200000
12	10 Best Foods for You	Thank you! Great app!! Add arthritis, eyes, im...	Positive	0.750000
13	10 Best Foods for You	Greatest ever Completely awesome maintain heal...	Positive	0.992188
14	10 Best Foods for You	Good health...... Good health first priority.....	Positive	0.550000
16	10 Best Foods for You	Health It's important world either life . thin...	Positive	0.450000
17	10 Best Foods for You	Mrs sunita bhati I thankful developers,to make...	Positive	0.600000
18	10 Best Foods for You	Very Useful in diabetes age 30. I need control...	Positive	0.295000
19	10 Best Foods for You	One greatest apps.	Positive	1.000000
20	10 Best Foods for You	good nice	Positive	0.650000
21	10 Best Foods for You	Healthy Really helped	Positive	0.350000
22	10 Best Foods for You	God health	Neutral	0.000000
23	10 Best Foods for You	HEALTH SHOULD ALWAYS BE TOP PRIORITY. !!. ON M...	Positive	0.781250
24	10 Best Foods for You	An excellent A useful	Positive	0.650000
25	10 Best Foods for You	I found lot wealth form health...	Neutral	0.000000
26	10 Best Foods for You	Because I found important.	Positive	0.400000
27	10 Best Foods for You	Healthy Eating	Positive	0.500000
28	10 Best Foods for You	Very good Simply good	Positive	0.805000
29	10 Best Foods for You	On test....	Neutral	0.000000
30	10 Best Foods for You	Good.!!	Positive	1.000000

	App	Review	Sentiment Category	Sentiment Score
31	10 Best Foods for You	Thanks advice. Downloaded Adobe reader still c...	Positive	0.200000
32	10 Best Foods for You	No recipe book Unable recipe book.	Negative	-0.500000
...	...	...	...	...
64185	Hotwire Hotel & Car Rental App	I hate gives notifications can't disabled. Thi...	Negative	-0.166667
64186	Hotwire Hotel & Car Rental App	Solid app. Booked many hotels & usually saved ...	Positive	0.165000
64187	Hotwire Hotel & Car Rental App	Always fair prices	Positive	0.700000
64188	Hotwire Hotel & Car Rental App	It work. It keeps saying problems try later. I...	Neutral	0.000000
64189	Hotwire Hotel & Car Rental App	Solid app. No issues yet. Been using year.	Neutral	0.000000
64190	Hotwire Hotel & Car Rental App	Find good price car rental.	Positive	0.700000
64191	Hotwire Hotel & Car Rental App	Very simple, straight forward, good ui....even...	Positive	0.350000
64192	Hotwire Hotel & Car Rental App	Great deals	Positive	0.800000
64194	Hotwire Hotel & Car Rental App	Keeps telling there's problem date time, can't...	Neutral	0.000000
64196	Housing-Real Estate & Property	Incorrect listings. The agents show property d...	Negative	-0.025000
64197	Housing-Real Estate & Property	Pathetic app.All posted leads owner either old...	Negative	-0.362500
64198	Housing-Real Estate & Property	It's waste app... Most properties 2017... And ...	Positive	0.266667
64199	Housing-Real Estate & Property	Filters useless I posted owners shows agents a...	Negative	-0.500000
64200	Housing-Real Estate & Property	Why business? Your filters don't work.. redire...	Neutral	0.000000
64201	Housing-Real Estate & Property	Filters work... Searching property useless	Negative	-0.500000
64205	Housing-Real Estate & Property	I want list property sale option. Please connect	Neutral	0.000000
64207	Housing-Real Estate & Property	Good	Positive	0.700000
64209	Housing-Real Estate & Property	Awesome app.. But prices date? Awesome app. Gi...	Positive	0.509375
64213	Housing-Real Estate & Property	This worse ever. They give graphic pictures pr...	Negative	-0.081250
64215	Housing-Real Estate & Property	Horrible app. I wanted list property get aroun...	Negative	-0.528571
64216	Housing-Real Estate & Property	Worst app. We get nothing Time waste . They up...	Negative	-0.400000

	App	Review	Sentiment Category	Sentiment Score
64217	Housing-Real Estate & Property	I able set range 1cr, scroll space 0-1cr range...	Positive	0.233333
64218	Housing-Real Estate & Property	What nonsensical app.. doesn't owner postings ...	Neutral	0.000000
64220	Housing-Real Estate & Property	No response support team. After I login, unabl...	Negative	-0.377778
64221	Housing-Real Estate & Property	Everything old stuff neither clear sold proper...	Positive	0.021591
64222	Housing-Real Estate & Property	Most ads older many agents ..not much owner po...	Positive	0.173333
64223	Housing-Real Estate & Property	If photos posted portal load, fit purpose. I'm...	Positive	0.225000
64226	Housing-Real Estate & Property	Dumb app, I wanted post property rent give opt...	Negative	-0.287500
64227	Housing-Real Estate & Property	I property business got link SMS happy perform...	Positive	0.800000
64230	Housing-Real Estate & Property	Useless app, I searched flats kondapur, Hydera...	Negative	-0.316667

37427 rows × 4 columns

In [23]:

```
#define data frame where apps are under category FINANCE
#from apps dataframe
finance_apps = apps[apps['Category'] == 'FINANCE']

#define data frame where apps are 'Free' under category 'Type
#from finance_apps dataframe
finance_apps_free = finance_apps[finance_apps['Type'] == 'Free']

#create a list from 'App' column in finance_apps_free
finance_apps_free_list = finance_apps_free['App'].tolist()

#verify list
print(finance_apps_free_list)
```

```
['K PLUS', 'ING Banking', 'Citibanamex Movil', 'The postal bank', 'KTB
Netbank', 'Mobile Bancomer', 'Nedbank Money', 'SCB EASY', 'CASHIER', 'R
abo Banking', 'Capitec Remote Banking', 'Itau bank', 'Nubank', 'The Soc
iete Generale App', 'IKO', 'Cash App', 'Standard Bank / Stanbic Bank',
'Bualuang mBanking', 'Intesa Sanpaolo Mobile', 'UBA Mobile Banking', 'B
BVA Spain', 'MyMo by GSB', 'VTB-Online', 'Ecobank Mobile Banking', 'Ban
orte Movil', 'Wells Fargo Mobile', 'Credit Karma', 'BZWBK24 mobile', 'P
ayPal', 'Capital One® Mobile', 'Zenith Bank Mobile App', 'GCash - Buy L
oad, Pay Bills, Send Money', 'Post Bank', 'İşCep', "People's Bank", 'Go
ogle Pay', 'Transfer', 'T-Mobile in', 'TrueMoney Wallet', 'Alfa-Bank (A
lfa-Bank)', 'Bank of Brazil', 'WiseBanyan - Invest For Free', 'Robinhoo
d - Investing, No Fees', 'Wells Fargo Daily Change', 'Even - organize y
our money, get paid early', 'Digit Save Money Automatically', 'Stash: I
nvest. Learn. Save.', 'Acorns - Invest Spare Change', 'Money Lover: Exp
ense Tracker, Budget Planner', 'Expense IQ Money Manager', 'Money Manag
er Expense & Budget', 'Prism Pay Bills, Track Money, Personal Finance',
'Monefy - Money Manager', 'Simple - Better Banking', 'Chime - Mobile Ba
nking', 'Bluebird by American Express', 'Walmart MoneyCard', 'Moven - S
mart Finances', 'Qapital - Save Small. Live Large', 'Experian - Free Cr
edit Report', 'Branch', 'WalletHub - Free Credit Score, Report & Monito
ring', 'CreditWise from Capital One', 'NerdWallet: Personal Finance, Cr
edit Score & Cash', 'my face', 'Credit Sesame', 'Mint: Budget, Bills, F
inance', 'Fresh EBT - Food Stamp Balance', 'Hurdlr: Track Mileage, Expe
nses, and Log Receipts', 'Mobills: Budget Planner', 'Everlance: Free Mi
leage Log', 'MileIQ - Free Mileage Tracker for Business', 'QuickBooks S
elf-Employed:Mileage Tracker and Taxes', 'FREE Stock Market Trading Tip
s', 'MetaTrader 5', 'MetaTrader 4', 'Stock Quote', 'Seeking Alpha', 'St
ock Trainer: Virtual Trading (Stock Markets)', 'Stocks: Realtime Quotes
Charts', 'Stocks, Forex, Bitcoin, Ethereum: Portfolio & News', 'CNBC: B
reaking Business News & Live Market Data', 'US Stock Market', 'Bloomber
g Professional', 'E*TRADE Mobile', 'MSN Money- Stock Quotes & News', 'F
ox Business', 'NSE Mobile Trading', 'Yahoo Finance', 'JStock - Stock Ma
rket, Portfolio & News', 'Webull - Stock Quotes & Free Stock Trading',
'Trading 212 - Forex, Stocks, CFDs', 'Moneycontrol – Stocks, Sensex, Mu
tual Funds, IPO', 'Current debit card and app made for teens', 'Betterm
ent', 'Citizens Bank Mobile Banking', 'GoBank', 'NetSpend Prepaid', 'BB
VA Compass Banking', 'BankMobile Vibe App', 'Netspend Skylight ONE', 'A
CE Elite', 'U.S. Bank', 'Huntington Mobile', 'USAA Mobile', 'Associated
Credit Union Mobile', 'Bank of America Mobile Banking', 'USE Credit Uni
on Mobile', 'Discover Mobile', 'Amex Mobile', 'Citi Mobile®', 'Navy Fed
eral Credit Union', 'Chase Mobile', 'HDFC Bank MobileBanking', 'Barclay
s US for Android', 'A+ Mobile', 'Fortune City - A Finance App', 'B', 'E
-NUM', 'Synd e-Passbook', 'F & M Mobile Banking', 'm-Faisaa', 'Q Mobile
Banking', 'R Bank', 'T. Rowe Price Personal® App', 'U by BB&T', 'U of I
Community Credit Union', 'AB Anywhere Mobile Banking', 'Hesab.af - Send
money in Afghanistan', 'EZ Ag Mobile', 'Alabama Ag Credit Ag Banking',
'Tri-Ag (WV) FCU', 'AJ Bell Youinvest', 'loans.com.au Smart Money', 'mo
n guide au bank populaire -infos et instructions', 'App Seguridad AV Vi
llas', 'AX Battery Saver', 'NB|AZ Mobile Banking', 'NB|AZ Business Mobi
le Banking', 'First Credit Union (AZ) Mobile', 'BD Bank Interest', 'Ear
n Money BD', 'be Produbanco', 'ePay.bg', 'BH Online', 'BH INVEST', 'Zir
aatBank BH', 'Bi en Línea', 'BI Mobile Banking', 'BI APP', 'BI Mobile',
'Capital Bk Mobile Business Dep', 'BK Gold App', 'BL Banking DES', 'BL
Mobile Banking', 'B L Enterprises', 'BM Wallet', 'bm Wallet', 'Mobiligh
t-BM', 'BM Assets', 'BN Más Cerca de Usted', 'BNCR Token Celular', 'BN
Bank Mobilbank', 'FX LITE BO', 'B R Telco FCU Mobile Banking', 'Banco S
abadell App. Your mobile bank', 'BT Panorama', 'Meu Cartão BV', 'Standa
rd Chartered Mobile (BW)', 'BW Mobilbanking für Smartphone und Tablet',
'Bitcoin BX Thailand', 'Bx Thailand Alert (Bitcoin)', 'Bitcoin Bx (Thai
land)', 'BX Thailand - Crypto Tracking & Analyse Bitcoin', 'BxPort - Bi
tcoin Bx (Thailand)', 'bX Thailand Simple Market Price and Alert', 'BX
Diff - Crypto Coins Checker', 'bitcoin exchange bx thailand', 'Bitcoin
& Cryptocurrency - Bx', 'Casa de Bolsa Bx+ Móvil', 'Crypto Tracker by B
itScreener', 'CoinMarketApp - CryptoCurrency Portfolio, News,ICO', 'Bit
coin Ticker Widget', 'Inwestor mobile', 'iBiznes24 mobile', 'CB-Mobile
Banking', 'CB Bank Mobile Banking', 'Citizens Bank - CB Mobile', 'CB Mo
bile Access', 'CB Trader', 'CB Pay', 'CB Mobile GA', 'CB Mobile Biz',
'C-E Federal Credit Union', 'CF Invest', 'CF PD', 'CF', '1st Fed CI Mob
ile Banking', 'CI Capital - Dynamic', 'CL e-bank', 'AND CO', 'REI Co-op
Mastercard', 'Co-op Credit Union on the Go', 'Sprig by CO-OP', 'Central
Bank and Trust Co.', 'Soo Co-op Mobile Banking', 'Co-Optima Mobile', 'G
renada Co-operative Bank', 'CP Federal Mobile', 'CQ Mobile', 'CQ SIGNAL
PRO 5', 'CT Checkout', 'CU: UPI Payments, Chat & Call', 'CU of Colorado
Mobile Banking', 'Tech CU Mobile Banking', 'Service CU Mobile Banking',
'Redwood CU v3.1.0', 'Cyprus CU Mobile Banking', 'BayPort CU Mobile Ban
king', 'CU SoCal Mobile Banking', 'Tropical Financial CU', 'Travis CU',
'Commonwealth CU Go Mobile', 'Public Service CU Mobile', 'Freedom CU',
'Meritrust CU Mobile Banking', 'Nusenda CU— Mobile Banking', 'Lake Mich
igan CU Mobile', 'Maps CU', 'Family Savings CU Mobile', 'FORUM Credit U
nion CU Online', 'Educators CU Mobile Banking', 'Space Coast CU Mobil
e', 'First Tech Federal CU', 'Southland CU', 'Tech CU Card Manager', 'N
umerica CU', 'Red River CU', 'Firefighters First CU', 'MYCU TX Mobile B
anking', 'Cy-Fair FCU Mobile Banking', 'CY.SEND Online Top Up', 'Alpha
Bank CY', 'EUROBANK CY', 'Rewards', 'Bank Of Cyprus', 'Diario Financier
o', 'Free coupons and vouchers', 'NY mobilbank DK - Danske Bank', 'D.L.
Evans Bank Mobile Banking', 'DM Accounting and Payroll', 'DN Calculator
s', 'DN Prasad', 'dt Pro', 'Direct Express®', 'CCP DZ : Fill out a chec
k | Number in letters DZD', 'Currency Exchange DZ', 'Devise Dz', 'Chang
eDA - Cours du DZD sur le marché parallèle', 'Tunisian Dinar: Exchange
rate', 'Exchange Bank - EB Mobile', 'EB-Banking (ersetzt)', 'Tamilnadu
EB Online Payment', 'Online Kerala EB Bill Pay', 'TNEB Quick Pay Easy',
'TNEB Bill Checker / TNEB Bill Status', 'EC Tax', 'EF-myHR', 'Education
First FCU Mobile', 'Financial Calculator Pro EF', 'EF Financial Control
Free', 'Bee Mobile EG', 'EG Tax Service', 'Alipay', 'EH National Mobile
Banking', 'EI Mobile', 'EI SmartMiles', 'Tigo Money El Salvador', 'The
dollar in mexico', 'Western Union US - Send Money Transfers Quickly',
'EO.TRADE - Coin sale', 'Cryptocurrency Trading - How To Trade Crypto',
'EP FCU', "Lead'er", 'ET Markets : NSE & BSE India', 'EU Mobile Money',
'EU Mobile Money Partenaire Commercial', 'Intra- and extra-EU trade dat
a', 'E W Bookkeeping & Accountancy', 'Money Manager Ex for Android', 'E
Y TaxChat', 'EY Expenses', 'EY Forensics', 'EZ Receipts', 'F&M Bank - E
Z Banking', 'EZ Financial Calculator', 'LEADS FA', 'Bank FD Interest Ca
lculator', 'FD Community FCU Mobile', 'Policy And FD Manager', 'FD Calc
ulator | Term Deposit | Fixed Deposit', 'Financial Calculator India',
'FD Calculator (EMI, SIP, RD & Loan Eligibility)', 'EMI, FD, RD - Ban
k Calculator', 'RD/FD Calculator', 'Fix Deposit Calculator (FD)', 'FD I
nterest Calculator', 'Deposit Calculator FD & RD', 'Banking Calculato
r', 'EMI/FD/RD/PV/IRR/BEP/EOQ Calc', 'EMI Calculator | FD Calculator |
RD calculator', 'Santa Fe FCU Mobile Banking', 'FE CREDIT - TET VUI',
'Caprock Santa Fe Credit Union', 'FE CPS', 'FG Wallet', 'FH Wallet', 'F
irst Hawaiian Bank Mobile', 'FI CFL', 'PriorityONE Credit Union of Fl',
'First Federal Bank of Florida', 'SB · FN 1870 Mobile Banking', 'FO BOU
LANGER', 'BankNordik', 'FP Markets', 'FP Boss', 'FP FCU', 'Gold Quote -
Gold.fr']
```

In [24]:

```
#group user_review dataframe by app name
#*Note: I had to include 'App' in the agg function otherwise code will not
#run. The liklihood that there is a better method to perform the following
#is possible.
user_reviews_grouped = user_reviews.groupby('App').agg({
            'App': 'count', 'Sentiment Score': 'mean'})

#user_reviews goruped returns App as an index, App as the count of apps
#grouped, and sentiment score
#rename App (count) to App_count so it may be dropped
user_reviews_grouped.rename(columns = {'App': 'App_count'}, inplace = True)

#remove App_count from the dataframe
user_reviews_grouped.drop(columns = 'App_count', inplace = True)

#remove index App so App becomes a column with app names
user_reviews_grouped.reset_index(level = 'App', inplace = True)

print(user_reviews_grouped.head())
```

```
                               App  Sentiment Score
0               10 Best Foods for You         0.470733
1  104 找工作 - 找工作 找打工 找兼職 履歷健檢 履歷診療室         0.392405
2                              11st         0.185943
3          1800 Contacts - Lens Store         0.318145
4   1LINE – One Line with One Touch         0.196290
```

In [25]:

```
#using the 'finance_apps_free_list' list previously created
#filter user_reviews for apps on the 'finance_apps_free_list'
free_finance_apps_reviews = user_reviews_grouped[user_reviews_grouped[
    'App'].isin(finance_apps_free_list)]

#Per requirements, sort by descending and take the top ten
#(first ten) rows
top_10_user_feedback = free_finance_apps_reviews.sort_values(
            'Sentiment Score', ascending = False)[0:10]
```

In [26]:

```
%%nose
# %%nose needs to be included at the beginning of every @tests cell

# https://instructor-support.datacamp.com/en/articles/4544008-writing-project-tests-guided-and-unguided-r-and-python
# The @solution should pass the tests
# The purpose of the tests is to try to catch common errors and
# to give the student a hint on how to resolve these errors

import numpy as np

correct_apps = pd.read_csv('datasets/apps.csv')
correct_reviews = pd.read_csv('datasets/user_reviews.csv')

# List of characters to remove
chars_to_remove = ['+', ',']
# Replace each character with an empty string
for char in chars_to_remove:
    correct_apps['Installs'] = correct_apps['Installs'].apply(lambda x: x.replace(char, ''))
# Convert col to int
correct_apps['Installs'] = correct_apps['Installs'].astype(int)


def test_pandas_loaded():
    assert ('pandas' in globals() or 'pd' in globals()), "pandas is not imported."

def test_installs_plus():
    assert '+' not in apps['Installs'], \
    'The special character "+" has not been removed from Installs column.'

def test_installs_comma():
    assert ',' not in apps['Installs'], \
    'The special character "," has not been removed from the Installs column.'

def test_installs_numeric():
    assert isinstance(apps['Installs'][0], np.int64), \
    'The Installs column is not of numeric data type (int).'

def test_q1_app_category_info_columns():

    # when DataFrame in MultiIndex
    if 'BEAUTY' in app_category_info.index:
        assert all(x in app_category_info.columns for x in ['Number of apps', 'Average price', 'Average rating']), \
        "Some columns are missing or incorrectly named in your app_category_info DataFrame. Make sure there are 4 columns named: 'Category', 'Number of apps', 'Average price', 'Average rating'."
    else:
        "Some columns are missing or incorrectly named in your app_category_info DataFrame. Make sure there are 4 columns named: 'Category', 'Number of apps', 'Average price', 'Average rating'."
```

```
def test_q1_app_category_info_app_count():

    if 'Number of apps' in app_category_info.reset_index().columns:
        correct_app_category_info = correct_apps.groupby(['Category']).a
gg({'App':'count', 'Price': 'mean', 'Rating': 'mean'}).reset_index()
        correct_app_category_info = correct_app_category_info.rename(col
umns={"App": "Number of apps", "Price": "Average price", "Rating": "Aver
age rating"})
        correct_app_count = correct_app_category_info['Number of apps']

        # convert to single index and compare
        app_count = app_category_info.reset_index().sort_values(by='Cate
gory')['Number of apps']
        assert correct_app_count.equals(app_count),\
        "The aggregate function used to calculate \"Number of apps\" is
 incorrect."

    else:
        assert False, "\"Number of apps\" column is missing in your app_
category_info DataFrame."


def test_q1_app_category_info_avg_price():

    if 'Average price' in app_category_info.reset_index().columns:
        correct_app_category_info = correct_apps.groupby(['Category']).a
gg({'App':'count', 'Price': 'mean', 'Rating': 'mean'}).reset_index()
        correct_app_category_info = correct_app_category_info.rename(col
umns={"App": "Number of apps", "Price": "Average price", "Rating": "Aver
age rating"})
        correct_app_count = correct_app_category_info['Average price']

        # convert to single index and compare
        app_count = app_category_info.reset_index().sort_values(by='Cate
gory')['Average price']
        assert correct_app_count.equals(app_count),\
        "The aggregate function used to calculate \"Average price\" is i
ncorrect."

    else:
        assert False, "\"Average price\" column is missing in your app_c
ategory_info DataFrame."

def test_q1_app_category_info_avg_rating():

    if 'Average rating' in app_category_info.reset_index().columns:
        correct_app_category_info = correct_apps.groupby('Category').agg
({'App':'count', 'Price': 'mean', 'Rating': 'mean'}).reset_index()
        correct_app_category_info = correct_app_category_info.rename(col
umns={"App": "Number of apps", "Price": "Average price", "Rating": "Aver
age rating"})
        correct_app_count = correct_app_category_info['Average rating']

        # convert to single index and compare
        app_count = app_category_info.reset_index().sort_values(by='Cate
gory')['Average rating']
        assert correct_app_count.equals(app_count),\
        "The aggregate function used to calculate \"Average rating\" is
 incorrect."

    else:
        assert False, "\"Average rating\" column is missing in your app_
category_info DataFrame."

# def test_reviews_loaded():
#     assert (correct_reviews.equals(reviews)), "The dataset was not rea
d correctly into reviews."

def test_q2_finance_apps():
    correct_finance_apps = correct_apps[(correct_apps['Type'] == 'Free')
& (correct_apps['Category'] == 'FINANCE')]['App']

    # if App column is the index
    if top_10_user_feedback.index.name == 'App':
        finance_apps = top_10_user_feedback.index
        assert(set(finance_apps).issubset(set(correct_finance_apps))),\
        "You have not selected the free finance apps correctly. Check yo
ur answer again."
    else:
        finance_apps = top_10_user_feedback['App']
        assert(set(finance_apps).issubset(set(correct_finance_apps))),\
        "You have not selected the free finance apps correctly. Check yo
ur answer again."


def test_q2_top_10():
    assert(len(top_10_user_feedback) == 10), "You have selected more tha
n 10 apps. Please select only top 10 apps with highest average sentiment
score."


def test_q2_sorted():
    correct_finance_apps = correct_apps[(correct_apps['Type'] == 'Free')
& (correct_apps['Category'] == 'FINANCE')]
    correct_merged_df = pd.merge(correct_finance_apps, correct_reviews,
 on = "App", how = "inner")

    correct_app_sentiment_score = correct_merged_df.groupby('App').agg
({'Sentiment Score': 'mean'}).reset_index()
    correct_sorted_apps = correct_app_sentiment_score.sort_values(by =
 'Sentiment Score', ascending = False)[:10]

    # if App column is the index
    if top_10_user_feedback.index.name == 'App':
        sorted_apps = top_10_user_feedback.index
        assert(list(sorted_apps) == list(correct_sorted_apps['App'])),\
        "You have not sorted top_10_user_feedback correctly. Make sure t
o sort your DataFrame on Sentiment Score from highest to lowest (ie - in
decreasing order)."
    else:
        sorted_apps = top_10_user_feedback['App']
        assert(list(sorted_apps) == list(correct_sorted_apps['App'])),\
        "You have not sorted top_10_user_feedback correctly. Make sure t
o sort your DataFrame on Sentiment Score from highest to lowest (ie - in
decreasing order)."


def test_q2():

    correct_finance_apps = correct_apps[(correct_apps['Type'] == 'Free')
& (correct_apps['Category'] == 'FINANCE')]
    correct_merged_df = pd.merge(correct_finance_apps, correct_reviews,
 on = "App", how = "inner")

    correct_app_sentiment_score = correct_merged_df.groupby('App').agg
({'Sentiment Score': 'mean'}).reset_index()
    correct_top_10_user_feedback = correct_app_sentiment_score.sort_valu
es(by = 'Sentiment Score', ascending = False).reset_index()[:10]

    correct_app_sentiment_score_multiindex = correct_merged_df.groupby
('App').agg({'Sentiment Score': 'mean'})
    correct_top_10_user_feedback_multiindex = correct_app_sentiment_scor
e_multiindex.sort_values(by = 'Sentiment Score', ascending = False)[:10]

    # if App column is the index
    if top_10_user_feedback.index.name == 'App':
        assert (correct_top_10_user_feedback_multiindex.equals(top_10_us
er_feedback)), "You have not computed top_10_user_feedback correctly. So
me values are wrong."
    else:
        top_10_user_feedback_apps = top_10_user_feedback['App']
        top_10_user_feedback_sentiment_score = top_10_user_feedback['Sen
timent Score']
        assert (list(top_10_user_feedback_apps) == list(correct_top_10_u
ser_feedback['App']) and
               list(top_10_user_feedback_sentiment_score) == list(correc
t_top_10_user_feedback['Sentiment Score'])), "You have not computed top_
10_user_feedback correctly. Some values are wrong."
```

```
Out[26]: 12/12 tests passed
```