Venkatasubramanian Narasimman
MS in Information Management
University of Illinois at Urbana Champaign

Fetch Rewards Data Analytics Internship Take Home Assessment
Exercise 2 – Answering questions from a business stakeholder

Dialect of SQL used: MySQL
Platform – MySQL Workbench

Notes:

Difficulty faced with data:
Some data points had double quotes in their values. This caused errors in insert statements. In order to resolve this, the double quotes in the data points were removed.

Also, please note that date attributes were created as VARCHAR during table creation and hence the STR_TO_DATE function has been used in the queries.

1. **Which brand saw the most dollars spent in the month of June?**
   - ➢ KIRKLAND SIGNATURE saw the most dollars spent in the month of June (2610.67) Top 10 brands have been printed in the result

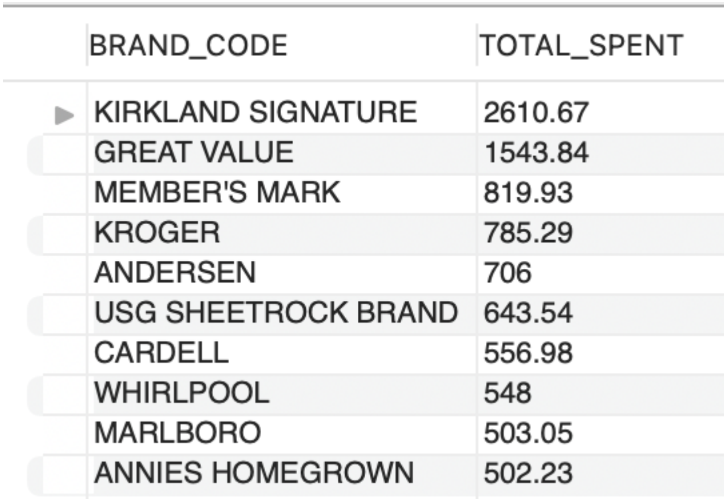
Query:

```
SELECT BRAND_CODE, ROUND(SUM(TOTAL_FINAL_PRICE),2) as TOTAL_SPENT
FROM RECEIPT_ITEMS
WHERE MONTH(STR_TO_DATE(MODIFY_DATE, '%Y-%m-%dT%H:%i:%s.%fZ')) = 6 AND
BRAND_CODE IS NOT NULL AND BRAND_CODE not like ''
GROUP BY BRAND_CODE
ORDER BY total_spent desc
LIMIT 10;
```

Query notes: Date is being stored as VARCHAR in the database as the given format in the csv file is string. To obtain the date, STR_TO_DATE is performed in the query the format present in the csv file. (This is being followed in the other queries involving date).

The brand code and the sum of the total final price was computed to see which brand saw the most dollars spent on it. Certain rows did not have a brand code and those rows were eliminated. The data was also filtered based on the month and only transactions during the month of June were considered. The data was grouped by the brand code and the result is sorted by the total amount spent.

Output



BRAND_CODE	TOTAL_SPENT
KIRKLAND SIGNATURE	2610.67
GREAT VALUE	1543.84
MEMBER'S MARK	819.93
KROGER	785.29
ANDERSEN	706
USG SHEETROCK BRAND	643.54
CARDELL	556.98
WHIRLPOOL	548
MARLBORO	503.05
ANNIES HOMEGROWN	502.23

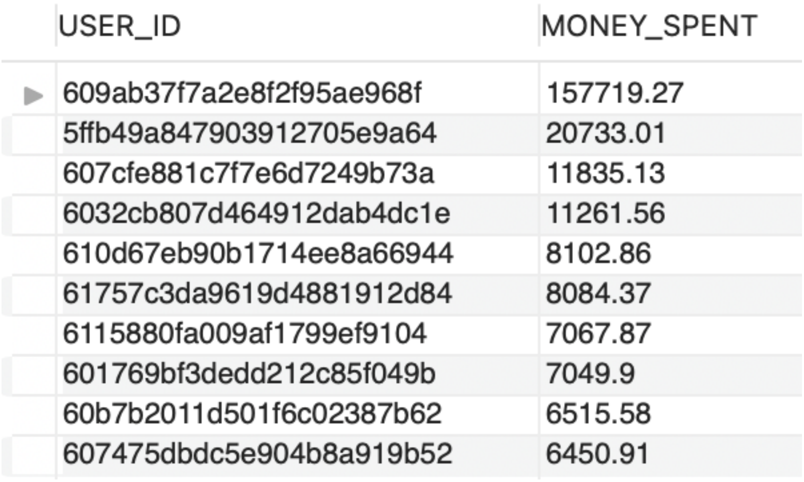
2. Which user spent the most money in the month of August?

- User with ID 609ab37f7a2e8f2f95ae968f spent 157719.27 in the month of August which is the highest amongst all users for that month.
  Top 10 results have been displayed.

Query

```
SELECT USER_ID, ROUND(SUM(TOTAL_SPENT),2) as MONEY_SPENT
FROM RECEIPTS
WHERE MONTH(STR_TO_DATE(MODIFY_DATE, '%Y-%m-%dT%H:%i:%s.%fZ')) = 8
GROUP BY USER_ID
ORDER BY MONEY_SPENT DESC
LIMIT 10;
```

Output



USER_ID	MONEY_SPENT
609ab37f7a2e8f2f95ae968f	157719.27
5ffb49a847903912705e9a64	20733.01
607cfe881c7f7e6d7249b73a	11835.13
6032cb807d464912dab4dc1e	11261.56
610d67eb90b1714ee8a66944	8102.86
61757c3da9619d4881912d84	8084.37
6115880fa009af1799ef9104	7067.87
601769bf3dedd212c85f049b	7049.9
60b7b2011d501f6c02387b62	6515.58
607475dbdc5e904b8a919b52	6450.91

3. **Which user bought the most expensive item?**

   - 3 users (IDs mentioned in the result) have bought the most expensive item.

   Query

   WITH MAX_PRICE AS (SELECT REWARDS_RECEIPT_ID, DESCRIPTION, BRAND_CODE,TOTAL_FINAL_PRICE FROM RECEIPT_ITEMS WHERE TOTAL_FINAL_PRICE = (SELECT MAX(TOTAL_FINAL_PRICE) FROM RECEIPT_ITEMS WHERE QUANTITY_PURCHASED>0))

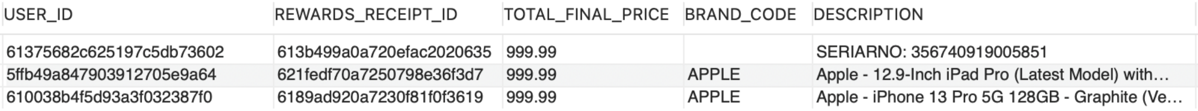
   SELECT r.USER_ID,mp.REWARDS_RECEIPT_ID, mp.TOTAL_FINAL_PRICE, mp.BRAND_CODE, DESCRIPTION FROM MAX_PRICE mp INNER JOIN receipts r ON mp.REWARDS_RECEIPT_ID = r.ID;

   Query notes: A common table expression MAX_PRICE is used to compute the most expensive item. A subquery returns the price of the most expensive item present in the receipts_item relation.

   This CTE is further joined with the receipts table to find the user ID. The common key used to join receipt and receipt_items is the ID and REWARDS_RECEIPT_ID of receipt and receipt_items respectively.

   Note that for certain items, the quantity purchased was null and this has been omitted by the condition that quantity purchased should be greater than 0.

   Output



USER_ID	REWARDS_RECEIPT_ID	TOTAL_FINAL_PRICE	BRAND_CODE	DESCRIPTION
61375682c625197c5db73602	613b499a0a720efac2020635	999.99		SERIARNO: 356740919005851
5ffb49a847903912705e9a64	621fedf70a7250798e36f3d7	999.99	APPLE	Apple - 12.9-Inch iPad Pro (Latest Model) with…
610038b4f5d93a3f032387f0	6189ad920a7230f81f0f3619	999.99	APPLE	Apple - iPhone 13 Pro 5G 128GB - Graphite (Ve…

4. What is the name of the most expensive item purchased?

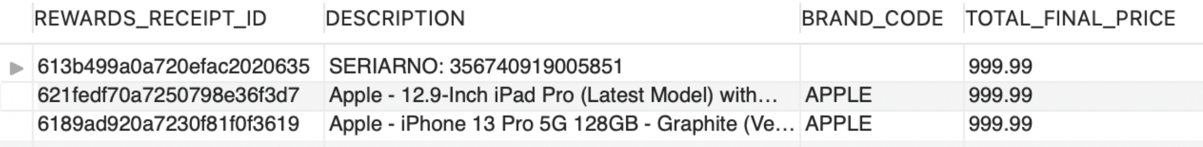
- 3 items were purchased for $ 999.99 as listed in the output.

Query

```
SELECT REWARDS_RECEIPT_ID, DESCRIPTION, BRAND_CODE,TOTAL_FINAL_PRICE
FROM RECEIPT_ITEMS WHERE TOTAL_FINAL_PRICE = (SELECT
MAX(TOTAL_FINAL_PRICE) FROM RECEIPT_ITEMS WHERE
QUANTITY_PURCHASED>0);
```

Query notes: A subquery returns the price of the most expensive item present in the receipt_items relation. This is further compared to all the rows in the same relation to find the most expensive item. 3 items were identified as the most expensive with the same price.

Output



REWARDS_RECEIPT_ID	DESCRIPTION	BRAND_CODE	TOTAL_FINAL_PRICE
613b499a0a720efac2020635	SERIARNO: 356740919005851		999.99
621fedf70a7250798e36f3d7	Apple - 12.9-Inch iPad Pro (Latest Model) with…	APPLE	999.99
6189ad920a7230f81f0f3619	Apple - iPhone 13 Pro 5G 128GB - Graphite (Ve…	APPLE	999.99

5. How many users scanned in each month?

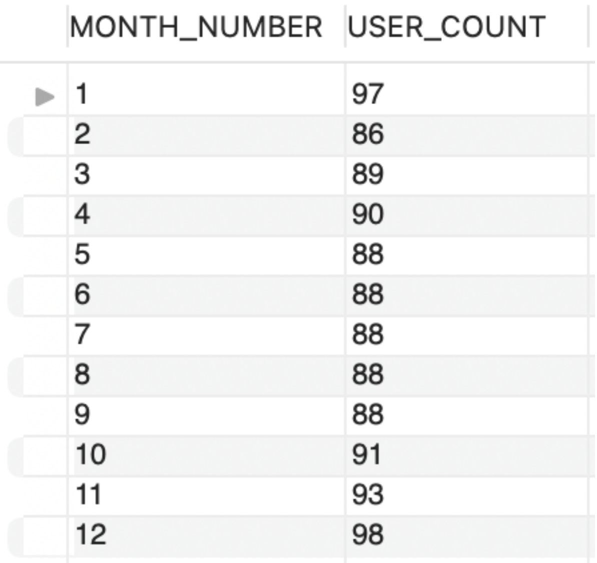
- The month number and the number of unique users who scanned during that particular month is computed.

Query

```
SELECT MONTH(STR_TO_DATE(DATE_SCANNED, '%Y-%m-%dT%H:%i:%s.%fZ')) AS
MONTH_NUMBER, COUNT(DISTINCT USER_ID) AS USER_COUNT
FROM RECEIPTS
WHERE MONTH(STR_TO_DATE(DATE_SCANNED, '%Y-%m-%dT%H:%i:%s.%fZ')) IS NOT
NULL
GROUP BY MONTH(STR_TO_DATE(DATE_SCANNED, '%Y-%m-%dT%H:%i:%s.%fZ'))
ORDER BY MONTH(STR_TO_DATE(DATE_SCANNED, '%Y-%m-%dT%H:%i:%s.%fZ'));
```

Query notes: The month number and the number of **unique** users who scanned in that particular month is computed by grouping the month number and counting the number of distinct users in each month. Some date columns were empty so the null columns were not considered in the computation.

Output



MONTH_NUMBER	USER_COUNT
1	97
2	86
3	89
4	90
5	88
6	88
7	88
8	88
9	88
10	91
11	93
12	98