coding_exercise_notebook

September 24, 2022

1 Coding Exercise Notebook

Created 8/24/2022 by

Timothy Del Green 1-256-335-0378 tdgreen@outlook.com https://www.linkedin.com/in/timothy-del-green

1.1 Installs and Imports

[109]:

```
# !pip install sqlalchemy
# !pip install pandas
# !pip install pandasql
# !pip install pandaserd
import numpy as np
import csv
import sqlite3
import pandas as pd
import json
import gzip
from pandaserd import ERD
from pandasql import sqldf
```

1.2 Receipts Data

[110]:

```
receipts_df = (
        pd.read_json(
            "https://fetch-hiring.s3.amazonaws.com/data-analyst/
 ↪ineeddata-data-modeling/receipts.json.gz",
            lines=True,
            compression='gzip'
        )
)

receipts_df.info()
receipts_df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1119 entries, 0 to 1118
```

```
Data columns (total 15 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   _id                      1119 non-null   object
 1   bonusPointsEarned        544 non-null    float64
 2   bonusPointsEarnedReason  544 non-null    object
 3   createDate               1119 non-null   object
 4   dateScanned              1119 non-null   object
 5   finishedDate             568 non-null    object
 6   modifyDate               1119 non-null   object
 7   pointsAwardedDate        537 non-null    object
 8   pointsEarned             609 non-null    float64
 9   purchaseDate             671 non-null    object
 10  purchasedItemCount       635 non-null    float64
 11  rewardsReceiptItemList   679 non-null    object
 12  rewardsReceiptStatus     1119 non-null   object
 13  totalSpent               684 non-null    float64
 14  userId                   1119 non-null   object
dtypes: float64(4), object(11)
memory usage: 131.3+ KB
```

```
[110]:                                        _id  bonusPointsEarned  \
       0  {'$oid': '5ff1e1eb0a720f0523000575'}              500.0
       1  {'$oid': '5ff1e1bb0a720f052300056b'}              150.0
       2  {'$oid': '5ff1e1f10a720f052300057a'}                5.0
       3  {'$oid': '5ff1e1ee0a7214ada100056f'}                5.0
       4  {'$oid': '5ff1e1d20a7214ada1000561'}                5.0

                                bonusPointsEarnedReason  \
       0  Receipt number 2 completed, bonus point schedu…
       1  Receipt number 5 completed, bonus point schedu…
       2                         All-receipts receipt bonus
       3                         All-receipts receipt bonus
       4                         All-receipts receipt bonus

                       createDate                dateScanned  \
       0  {'$date': 1609687531000}  {'$date': 1609687531000}
       1  {'$date': 1609687483000}  {'$date': 1609687483000}
       2  {'$date': 1609687537000}  {'$date': 1609687537000}
       3  {'$date': 1609687534000}  {'$date': 1609687534000}
       4  {'$date': 1609687506000}  {'$date': 1609687506000}

                     finishedDate                 modifyDate  \
       0  {'$date': 1609687531000}  {'$date': 1609687536000}
       1  {'$date': 1609687483000}  {'$date': 1609687488000}
       2                       NaN  {'$date': 1609687542000}
       3  {'$date': 1609687534000}  {'$date': 1609687539000}
```

```
4  {'$date': 1609687511000}  {'$date': 1609687511000}

          pointsAwardedDate  pointsEarned             purchaseDate  \
0  {'$date': 1609687531000}         500.0  {'$date': 1609632000000}
1  {'$date': 1609687483000}         150.0  {'$date': 1609601083000}
2                       NaN           5.0  {'$date': 1609632000000}
3  {'$date': 1609687534000}           5.0  {'$date': 1609632000000}
4  {'$date': 1609687506000}           5.0  {'$date': 1609601106000}

   purchasedItemCount                            rewardsReceiptItemList  \
0                 5.0  [{'barcode': '4011', 'description': 'ITEM NOT …
1                 2.0  [{'barcode': '4011', 'description': 'ITEM NOT …
2                 1.0  [{'needsFetchReview': False, 'partnerItemId': …
3                 4.0  [{'barcode': '4011', 'description': 'ITEM NOT …
4                 2.0  [{'barcode': '4011', 'description': 'ITEM NOT …

  rewardsReceiptStatus  totalSpent                    userId
0             FINISHED        26.0  5ff1e1eacfcf6c399c274ae6
1             FINISHED        11.0  5ff1e194b6a9d73a3a9f1052
2             REJECTED        10.0  5ff1e1f1cfcf6c399c274b0b
3             FINISHED        28.0  5ff1e1eacfcf6c399c274ae6
4             FINISHED         1.0  5ff1e194b6a9d73a3a9f1052
```

[111]:

```python
def explode_and_normalize_item_list(df_, to_explode):

    df_ = df_.explode(to_explode)

    df_ = pd.json_normalize(json.loads(df_.to_json(orient="records")))

    return df_
```

[112]:

```python
def receipts_table(df):
    return (
        df
            .assign(
                finishedDate = pd.json_normalize(df['finishedDate']),
                pointsAwardedDate = pd.json_normalize(df['pointsAwardedDate']),
                purchaseDate = pd.json_normalize(df['purchaseDate']),

                **{col : pd.json_normalize(df[col]) for col in [
                    '_id',
                    'createDate',
                    'dateScanned',
                    'modifyDate',
                ]},
            )
            .pipe(explode_and_normalize_item_list, 'rewardsReceiptItemList')
            .astype({
                **{col : str for col in [
                    'purchaseDate'
                ]}
            })
            .assign(
                # Extract first 10 characters for Unix timestamp
                purchaseDate = lambda df_ : df_['purchaseDate'].str.extract('(^\d{10})')
            )
            .assign(
                # Convert Unix timestamp to datetime
                purchaseDate = lambda df_ : pd.to_datetime(df_['purchaseDate'], unit='s')
            )
    )


receipts_table(receipts_df).info()
receipts_table(receipts_df).head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7381 entries, 0 to 7380
Data columns (total 49 columns):
 #   Column                                                     Non-Null Count
Dtype
---  ------                                                     --------------
-----
 0   _id                                                        7381 non-null
object
 1   bonusPointsEarned                                          5980 non-null
float64
 2   bonusPointsEarnedReason                                    5980 non-null
object
 3   createDate                                                 7381 non-null
int64
 4   dateScanned                                                7381 non-null
int64
 5   finishedDate                                               5970 non-null
float64
 6   modifyDate                                                 7381 non-null
int64
 7   pointsAwardedDate                                          6080 non-null
float64
 8   pointsEarned                                               6253 non-null
float64
 9   purchaseDate                                               6923 non-null
```

```
datetime64[ns]
 10  purchasedItemCount                                          6897 non-null
float64
 11  rewardsReceiptStatus                                        7381 non-null
object
 12  totalSpent                                                  6946 non-null
float64
 13  userId                                                      7381 non-null
object
 14  rewardsReceiptItemList.barcode                              3090 non-null
object
 15  rewardsReceiptItemList.description                          6560 non-null
object
 16  rewardsReceiptItemList.finalPrice                           6767 non-null
object
 17  rewardsReceiptItemList.itemPrice                            6767 non-null
object
 18  rewardsReceiptItemList.needsFetchReview                     813 non-null
object
 19  rewardsReceiptItemList.partnerItemId                        6941 non-null
object
 20  rewardsReceiptItemList.preventTargetGapPoints               358 non-null
object
 21  rewardsReceiptItemList.quantityPurchased                    6767 non-null
float64
 22  rewardsReceiptItemList.userFlaggedBarcode                   337 non-null
object
 23  rewardsReceiptItemList.userFlaggedNewItem                   323 non-null
object
 24  rewardsReceiptItemList.userFlaggedPrice                     299 non-null
object
 25  rewardsReceiptItemList.userFlaggedQuantity                  299 non-null
float64
 26  rewardsReceiptItemList.needsFetchReviewReason               219 non-null
object
 27  rewardsReceiptItemList.pointsNotAwardedReason               340 non-null
object
 28  rewardsReceiptItemList.pointsPayerId                        1267 non-null
object
 29  rewardsReceiptItemList.rewardsGroup                         1731 non-null
object
 30  rewardsReceiptItemList.rewardsProductPartnerId              2269 non-null
object
 31  rewardsReceiptItemList.userFlaggedDescription               205 non-null
object
 32  rewardsReceiptItemList.originalMetaBriteBarcode             71 non-null
object
 33  rewardsReceiptItemList.originalMetaBriteDescription         10 non-null
```

```
object
 34  rewardsReceiptItemList.brandCode                           2600 non-null
object
 35  rewardsReceiptItemList.competitorRewardsGroup              275 non-null
object
 36  rewardsReceiptItemList.discountedItemPrice                 5769 non-null
object
 37  rewardsReceiptItemList.originalReceiptItemText             5760 non-null
object
 38  rewardsReceiptItemList.itemNumber                          153 non-null
object
 39  rewardsReceiptItemList.originalMetaBriteQuantityPurchased  15 non-null
float64
 40  rewardsReceiptItemList.pointsEarned                        927 non-null
object
 41  rewardsReceiptItemList.targetPrice                         378 non-null
object
 42  rewardsReceiptItemList.competitiveProduct                  645 non-null
object
 43  rewardsReceiptItemList.originalFinalPrice                  9 non-null
object
 44  rewardsReceiptItemList.originalMetaBriteItemPrice          9 non-null
object
 45  rewardsReceiptItemList.deleted                             9 non-null
object
 46  rewardsReceiptItemList.priceAfterCoupon                    956 non-null
object
 47  rewardsReceiptItemList                                     0 non-null
float64
 48  rewardsReceiptItemList.metabriteCampaignId                 863 non-null
object
dtypes: datetime64[ns](1), float64(10), int64(3), object(35)
memory usage: 2.8+ MB
```

[112]:

```
                        _id  bonusPointsEarned  \
0  5ff1e1eb0a720f0523000575              500.0
1  5ff1e1bb0a720f052300056b              150.0
2  5ff1e1bb0a720f052300056b              150.0
3  5ff1e1f10a720f052300057a                5.0
4  5ff1e1ee0a7214ada100056f                5.0

                             bonusPointsEarnedReason     createDate  \
0  Receipt number 2 completed, bonus point schedu…  1609687531000
1  Receipt number 5 completed, bonus point schedu…  1609687483000
2  Receipt number 5 completed, bonus point schedu…  1609687483000
3                         All-receipts receipt bonus  1609687537000
4                         All-receipts receipt bonus  1609687534000
```

```
     dateScanned  finishedDate     modifyDate  pointsAwardedDate  \
0  1609687531000  1.609688e+12  1609687536000       1.609688e+12
1  1609687483000  1.609687e+12  1609687488000       1.609687e+12
2  1609687483000  1.609687e+12  1609687488000       1.609687e+12
3  1609687537000           NaN  1609687542000                NaN
4  1609687534000  1.609688e+12  1609687539000       1.609688e+12

   pointsEarned        purchaseDate  …  \
0         500.0 2021-01-03 00:00:00  …
1         150.0 2021-01-02 15:24:43  …
2         150.0 2021-01-02 15:24:43  …
3           5.0 2021-01-03 00:00:00  …
4           5.0 2021-01-03 00:00:00  …

   rewardsReceiptItemList.originalMetaBriteQuantityPurchased  \
0                                                NaN
1                                                NaN
2                                                NaN
3                                                NaN
4                                                NaN

  rewardsReceiptItemList.pointsEarned  rewardsReceiptItemList.targetPrice  \
0                                 NaN                                 NaN
1                                 NaN                                 NaN
2                                 NaN                                 NaN
3                                 NaN                                 NaN
4                                 NaN                                 NaN

  rewardsReceiptItemList.competitiveProduct  \
0                                       NaN
1                                       NaN
2                                       NaN
3                                       NaN
4                                       NaN

  rewardsReceiptItemList.originalFinalPrice  \
0                                       NaN
1                                       NaN
2                                       NaN
3                                       NaN
4                                       NaN

  rewardsReceiptItemList.originalMetaBriteItemPrice  \
0                                               NaN
1                                               NaN
2                                               NaN
```

```
3                                               NaN
4                                               NaN

  rewardsReceiptItemList.deleted rewardsReceiptItemList.priceAfterCoupon  \
0                            NaN                                     NaN
1                            NaN                                     NaN
2                            NaN                                     NaN
3                            NaN                                     NaN
4                            NaN                                     NaN

  rewardsReceiptItemList rewardsReceiptItemList.metabriteCampaignId
0                    NaN                                        NaN
1                    NaN                                        NaN
2                    NaN                                        NaN
3                    NaN                                        NaN
4                    NaN                                        NaN

[5 rows x 49 columns]
```

1.3 Users Table

[113]:
```
users_df = (
        pd.read_json(
            "https://fetch-hiring.s3.amazonaws.com/data-analyst/
 ↪ineeddata-data-modeling/users.json.gz",
            lines=True,
            compression='gzip'
        )
)

users_df.info()
users_df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 495 entries, 0 to 494
Data columns (total 7 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   _id           495 non-null    object
 1   active        495 non-null    bool
 2   createdDate   495 non-null    object
 3   lastLogin     433 non-null    object
 4   role          495 non-null    object
 5   signUpSource  447 non-null    object
 6   state         439 non-null    object
dtypes: bool(1), object(6)
memory usage: 23.8+ KB
```

```
[113]:                                     _id  active              createdDate  \
       0  {'$oid': '5ff1e194b6a9d73a3a9f1052'}    True  {'$date': 1609687444800}
       1  {'$oid': '5ff1e194b6a9d73a3a9f1052'}    True  {'$date': 1609687444800}
       2  {'$oid': '5ff1e194b6a9d73a3a9f1052'}    True  {'$date': 1609687444800}
       3  {'$oid': '5ff1e1eacfcf6c399c274ae6'}    True  {'$date': 1609687530554}
       4  {'$oid': '5ff1e194b6a9d73a3a9f1052'}    True  {'$date': 1609687444800}

                         lastLogin      role signUpSource state
       0  {'$date': 1609687537858}  consumer        Email    WI
       1  {'$date': 1609687537858}  consumer        Email    WI
       2  {'$date': 1609687537858}  consumer        Email    WI
       3  {'$date': 1609687530597}  consumer        Email    WI
       4  {'$date': 1609687537858}  consumer        Email    WI
```

```
[129]: def users_table(df):
           return (
               df
                   .assign(
                       lastLogin = pd.json_normalize(df['lastLogin']),

                       **{col : pd.json_normalize(df[col]) for col in [
                           '_id',
                           'createdDate',
                       ]},
                   )
                   .astype({
                       **{col : str for col in [
                           'createdDate'
                       ]}
                   })
                   .assign(
                       createdDate = lambda df_ : df_['createdDate'].str.
         ↪extract('(^\d{10})')
                   )
                   .assign(
                       createdDate = lambda df_ : pd.to_datetime(df_['createdDate'],␣
         ↪unit='s')
                   )
           )

       users_table(users_df).info()
       users_table(users_df).head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 495 entries, 0 to 494
Data columns (total 7 columns):
 #   Column        Non-Null Count  Dtype
```

```
---  ------         --------------  -----
 0   _id            495 non-null    object
 1   active         495 non-null    bool
 2   createdDate    495 non-null    datetime64[ns]
 3   lastLogin      433 non-null    float64
 4   role           495 non-null    object
 5   signUpSource   447 non-null    object
 6   state          439 non-null    object
dtypes: bool(1), datetime64[ns](1), float64(1), object(4)
memory usage: 23.8+ KB
```

```
[129]:                        _id  active         createdDate     lastLogin  \
        0  5ff1e194b6a9d73a3a9f1052    True 2021-01-03 15:24:04  1.609688e+12
        1  5ff1e194b6a9d73a3a9f1052    True 2021-01-03 15:24:04  1.609688e+12
        2  5ff1e194b6a9d73a3a9f1052    True 2021-01-03 15:24:04  1.609688e+12
        3  5ff1e1eacfcf6c399c274ae6    True 2021-01-03 15:25:30  1.609688e+12
        4  5ff1e194b6a9d73a3a9f1052    True 2021-01-03 15:24:04  1.609688e+12

               role signUpSource state
        0  consumer        Email    WI
        1  consumer        Email    WI
        2  consumer        Email    WI
        3  consumer        Email    WI
        4  consumer        Email    WI
```

1.4 Brand Table

```
[130]: brand_df = (
               pd.read_json(
                   "https://fetch-hiring.s3.amazonaws.com/data-analyst/
         ↪ineeddata-data-modeling/brands.json.gz",
                   lines=True,
                   compression='gzip'
               )
       )

       brand_df.info()
       brand_df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1167 entries, 0 to 1166
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   _id            1167 non-null   object
 1   barcode        1167 non-null   int64
 2   category       1012 non-null   object
 3   categoryCode  517 non-null    object
 4   cpg           1167 non-null   object
 5   name          1167 non-null   object
 6   topBrand      555 non-null    float64
 7   brandCode     933 non-null    object
dtypes: float64(1), int64(1), object(6)
memory usage: 73.1+ KB
```

[130]:

```
                                    _id       barcode        category  \
0  {'$oid': '601ac115be37ce2ead437551'}  511111019862          Baking
1  {'$oid': '601c5460be37ce2ead43755f'}  511111519928       Beverages
2  {'$oid': '601ac142be37ce2ead43755d'}  511111819905          Baking
3  {'$oid': '601ac142be37ce2ead43755a'}  511111519874          Baking
4  {'$oid': '601ac142be37ce2ead43755e'}  511111319917  Candy & Sweets

      categoryCode                                                cpg  \
0           BAKING  {'$id': {'$oid': '601ac114be37ce2ead437550'}, …
1        BEVERAGES  {'$id': {'$oid': '5332f5fbe4b03c9a25efd0ba'}, …
2           BAKING  {'$id': {'$oid': '601ac142be37ce2ead437559'}, …
3           BAKING  {'$id': {'$oid': '601ac142be37ce2ead437559'}, …
4  CANDY_AND_SWEETS  {'$id': {'$oid': '5332fa12e4b03c9a25efd1e7'}, …

                         name  topBrand                      brandCode
0  test brand @1612366101024       0.0                            NaN
1                   Starbucks       0.0                      STARBUCKS
2  test brand @1612366146176       0.0  TEST BRANDCODE @1612366146176
3  test brand @1612366146051       0.0  TEST BRANDCODE @1612366146051
4  test brand @1612366146827       0.0  TEST BRANDCODE @1612366146827
```

[131]:

```
def brand_table(df):
    return (
        df
            .assign(
                **{col : pd.json_normalize(df[col]) for col in [
                    '_id',
                ]},
            )
            .pipe(lambda df_ : pd.json_normalize(json.loads(df_.
 ↪to_json(orient="records"))))
    )

brand_table(brand_df).info()
brand_table(brand_df).head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1167 entries, 0 to 1166
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
```

```
---  ------        --------------  -----
 0   _id           1167 non-null   object
 1   barcode       1167 non-null   int64
 2   category      1012 non-null   object
 3   categoryCode  517 non-null    object
 4   name          1167 non-null   object
 5   topBrand      555 non-null    float64
 6   brandCode     933 non-null    object
 7   cpg.$id.$oid  1167 non-null   object
 8   cpg.$ref      1167 non-null   object
dtypes: float64(1), int64(1), object(7)
memory usage: 82.2+ KB
```

[131]:

```
                        _id       barcode        category      categoryCode  \
0  601ac115be37ce2ead437551  511111019862          Baking            BAKING
1  601c5460be37ce2ead43755f  511111519928       Beverages         BEVERAGES
2  601ac142be37ce2ead43755d  511111819905          Baking            BAKING
3  601ac142be37ce2ead43755a  511111519874          Baking            BAKING
4  601ac142be37ce2ead43755e  511111319917  Candy & Sweets  CANDY_AND_SWEETS

                       name  topBrand                      brandCode  \
0  test brand @1612366101024       0.0                           None
1                 Starbucks       0.0                      STARBUCKS
2  test brand @1612366146176       0.0  TEST BRANDCODE @1612366146176
3  test brand @1612366146051       0.0  TEST BRANDCODE @1612366146051
4  test brand @1612366146827       0.0  TEST BRANDCODE @1612366146827

               cpg.$id.$oid cpg.$ref
0  601ac114be37ce2ead437550     Cogs
1  5332f5fbe4b03c9a25efd0ba     Cogs
2  601ac142be37ce2ead437559     Cogs
3  601ac142be37ce2ead437559     Cogs
4  5332fa12e4b03c9a25efd1e7     Cogs
```

1.5 Create Tables to be Queried

[117]:

```
RECEIPTSTBL = receipts_table(receipts_df)
USERSTBL    = users_table(users_df)
BRANDTBL    = brand_table(brand_df)
```

1.6 Data Queries

1.6.1 What are the top 5 brands by receipts scanned for most recent month?

The most recent month in the receipts table is March, 2021. There is only one brand code with scanned receipts in that month, that brand code being ‘No Brand Code’ with 2 receipts scanned.

```
[121]: q_1 = """

WITH
    RECEIPT_TABLE AS (
        SELECT DISTINCT
            CASE
                WHEN A."rewardsReceiptItemList.brandCode" IS NULL
                    THEN "No Brand Code"
                ELSE A."rewardsReceiptItemList.brandCode"
            END             AS BRAND,

            STRFTIME('%m', DATE(A.purchaseDate)) AS PURCHASE_MONTH,

            STRFTIME('%Y', DATE(A.purchaseDate)) AS PURCHASE_YEAR,

            MAX(A.purchaseDate) OVER ()
                            AS MOST_RECENT_DATE,

            COUNT(A._id) OVER (
                PARTITION BY
                    A."rewardsReceiptItemList.brandCode",
                    STRFTIME('%Y', DATE(A.purchaseDate)),
                    STRFTIME('%m', DATE(A.purchaseDate))
                ORDER BY
                    A."rewardsReceiptItemList.brandCode",
                    CAST(STRFTIME('%Y', DATE(A.purchaseDate)) AS INTEGER),
                    CAST(STRFTIME('%m', DATE(A.purchaseDate)) AS INTEGER)

            )               AS RECEIPTS_SCANNED

        FROM RECEIPTSTBL AS A

        ORDER BY
            A."rewardsReceiptItemList.brandCode",
            STRFTIME('%Y', DATE(A.purchaseDate)),
            STRFTIME('%m', DATE(A.purchaseDate))
    )

SELECT
    A.BRAND,

    A.PURCHASE_YEAR,

    A.PURCHASE_MONTH,

    A.RECEIPTS_SCANNED,
    LAG(A.PURCHASE_MONTH) OVER (
        PARTITION BY
            A.BRAND
        ORDER BY
            A.PURCHASE_YEAR,
            A.PURCHASE_MONTH
    )                   AS PREVIOUS_PERIOD,

    LAG(A.RECEIPTS_SCANNED) OVER (
        PARTITION BY
            A.BRAND
        ORDER BY
            A.PURCHASE_YEAR,
            A.PURCHASE_MONTH
    )                   AS RECEIPTS_SCANNED_PREVIOUS_PERIOD

FROM RECEIPT_TABLE AS A

WHERE
        A.PURCHASE_MONTH >= STRFTIME('%m', DATE(A.MOST_RECENT_DATE, '-1 MONTH'))
    AND A.PURCHASE_YEAR = STRFTIME('%Y', DATE(A.MOST_RECENT_DATE))

ORDER BY
    A.PURCHASE_YEAR,
    A.PURCHASE_MONTH DESC

LIMIT 5

"""

sqldf(q_1, globals())
```

[121]:

```
           BRAND PURCHASE_YEAR PURCHASE_MONTH  RECEIPTS_SCANNED  \
0  No Brand Code          2021             03                 2
1          BRAND          2021             02                 1
2        MISSION          2021             02                 2
3  No Brand Code          2021             02               168
4           VIVA          2021             02                 1

  PREVIOUS_PERIOD  RECEIPTS_SCANNED_PREVIOUS_PERIOD
0              02                             168.0
1            None                               NaN
2            None                               NaN
3            None                               NaN
4            None                               NaN
```

1.6.2 How does the ranking of the top 5 brands by receipts scanned for the recent month compare to the ranking for the previous month?

“No Brand Code”, being the only brand to have associated receipts scanned in the most recent month, saw a decrease of 166 scanned receipts between February 2021 and March 2021.

[122]:
```
q_2 = """

WITH
    RECEIPT_TABLE AS (
        SELECT DISTINCT
            CASE
                WHEN A."rewardsReceiptItemList.brandCode" IS NULL
                    THEN "No Brand Code"
                ELSE A."rewardsReceiptItemList.brandCode"
            END             AS BRAND,

            STRFTIME('%m', DATE(A.purchaseDate)) AS PURCHASE_MONTH,

            STRFTIME('%Y', DATE(A.purchaseDate)) AS PURCHASE_YEAR,

            MAX(A.purchaseDate) OVER ()
                            AS MOST_RECENT_DATE,

            COUNT(A._id) OVER (
                PARTITION BY
                    A."rewardsReceiptItemList.brandCode",
                    STRFTIME('%Y', DATE(A.purchaseDate)),
                    STRFTIME('%m', DATE(A.purchaseDate))
                ORDER BY
                    A."rewardsReceiptItemList.brandCode",
                    CAST(STRFTIME('%Y', DATE(A.purchaseDate)) AS INTEGER),
                    CAST(STRFTIME('%m', DATE(A.purchaseDate)) AS INTEGER)

            )               AS RECEIPTS_SCANNED

        FROM RECEIPTSTBL AS A

        ORDER BY
            A."rewardsReceiptItemList.brandCode",
            STRFTIME('%Y', DATE(A.purchaseDate)),
            STRFTIME('%m', DATE(A.purchaseDate))
    )

SELECT
    A.BRAND,

    A.PURCHASE_YEAR,
    A.PURCHASE_MONTH,

    A.RECEIPTS_SCANNED  AS RECEIPTS_SCANNED,

    LAG(A.PURCHASE_MONTH) OVER (
        PARTITION BY
            A.BRAND
        ORDER BY
            A.PURCHASE_YEAR,
            A.PURCHASE_MONTH
    )                   AS PREVIOUS_PERIOD,

    LAG(A.RECEIPTS_SCANNED) OVER (
        PARTITION BY
            A.BRAND
        ORDER BY
            A.PURCHASE_YEAR,
            A.PURCHASE_MONTH
    )                   AS RECEIPTS_SCANNED_PREVIOUS_PERIOD

FROM RECEIPT_TABLE AS A

WHERE
        A.PURCHASE_MONTH >= STRFTIME('%m', DATE(A.MOST_RECENT_DATE, '-1 MONTH'))
    AND A.PURCHASE_YEAR = STRFTIME('%Y', DATE(A.MOST_RECENT_DATE))

ORDER BY
    A.PURCHASE_YEAR,
    A.PURCHASE_MONTH DESC,
    A.RECEIPTS_SCANNED DESC

"""

sqldf(q_2, globals())
```

[122]:

```
           BRAND PURCHASE_YEAR PURCHASE_MONTH  RECEIPTS_SCANNED  \
0  No Brand Code          2021             03                 2
1  No Brand Code          2021             02               168
2        MISSION          2021             02                 2
3          BRAND          2021             02                 1
4           VIVA          2021             02                 1

  PREVIOUS_PERIOD  RECEIPTS_SCANNED_PREVIOUS_PERIOD
0              02                             168.0
1            None                               NaN
2            None                               NaN
```

```
3            None                                  NaN
4            None                                  NaN
```

1.6.3 When considering average spend from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', whis is greater?

In the unique values for rewardsReceiptStatus, there is no value 'ACCEPTED'. So, 'REJECTED' would be greater by default, with an average value of $19.54.

[123]:

```
q_3 = """

SELECT
    rewardsReceiptStatus,
    AVG(totalSpent) AS AVG_SPENT

FROM
    RECEIPTSTBL AS A

GROUP BY
    rewardsReceiptStatus

"""

sqldf(q_3, globals())
```

[123]:

```
  rewardsReceiptStatus    AVG_SPENT
0             FINISHED  1244.372934
1              FLAGGED  2635.570247
2              PENDING    28.032449
3             REJECTED    19.544970
4            SUBMITTED          NaN
```

1.6.4 When considering total number of items purchased from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?

In the unique values for rewardsReceiptStatus, there is no value 'ACCEPTED'. So, 'REJECTED' would be greater by default, with a value of 167

[124]:

```
q_4 = """

SELECT
    rewardsReceiptStatus,
    COUNT(_id)

FROM
    RECEIPTSTBL AS A

```

```
GROUP BY
    rewardsReceiptStatus

"""

sqldf(q_4, globals())
```

[124]:

```
  rewardsReceiptStatus  COUNT(_id)
0             FINISHED        5920
1              FLAGGED         810
2              PENDING          50
3             REJECTED         167
4            SUBMITTED         434
```

1.6.5 Which brand has the most spend among users who were created within the past 6 months?

Here, we have used a time offset of 20 months in order to produce example results. The highest value belongs to “No Brand Code”, followed by Kroger with a value of $222,538.59

[125]:

```
q_5 = """

SELECT
    CASE
        WHEN A."rewardsReceiptItemList.brandCode" IS NULL
            THEN "No Brand Code"
        ELSE A."rewardsReceiptItemList.brandCode"
    END                 AS BRAND,

    SUM(A.totalSpent)   AS TOTAL_SPENT

FROM RECEIPTSTBL AS A
    JOIN USERSTBL AS B
        ON A.userID = b._id

WHERE
    B.createdDate >= DATE('NOW', '-20 MONTH')

GROUP BY
    A."rewardsReceiptItemList.brandCode"

ORDER BY
    SUM(A.totalSpent) DESC

LIMIT 5

"""
```

```

sqldf(q_5, globals())
```

[125]:

```
            BRAND  TOTAL_SPENT
0   No Brand Code   2561099.01
1          KROGER    222538.59
2  BEN AND JERRYS    153193.80
3        PRINGLES     62485.46
4           KRAFT     61032.20
```

1.6.6 Which brand has the most transactions among users who were created within the past 6 months?

Here, we have used a time offset of 20 months in order to produce example results. The highest value belongs to “No Brand Code”, followed by KROGER with a transaction count of 65.

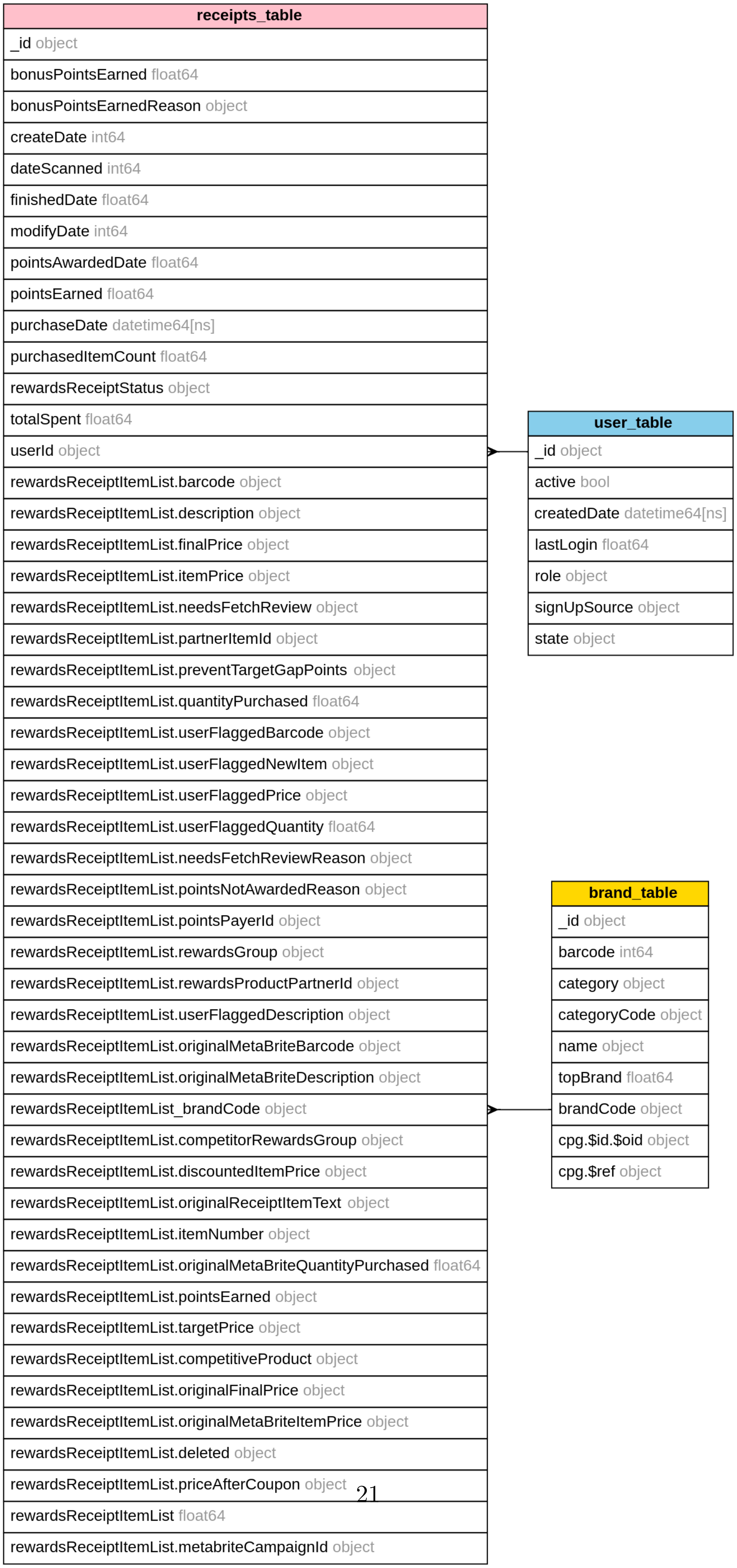
[126]:

```
q_6 = """

SELECT
    CASE
        WHEN A."rewardsReceiptItemList.brandCode" IS NULL
            THEN "No Brand Code"
        ELSE A."rewardsReceiptItemList.brandCode"
    END                 AS BRAND,

    COUNT(A._id)          AS COUNT_OF_TRANSACTIONS

FROM RECEIPTSTBL AS A
    JOIN USERSTBL AS B
        ON A.userID = b._id

WHERE
    B.createdDate >= DATE('NOW', '-20 MONTH')

GROUP BY
    A."rewardsReceiptItemList.brandCode"

ORDER BY
    COUNT(A._id) DESC

LIMIT 5

"""

sqldf(q_6, globals())
```

```
[126]:            BRAND  COUNT_OF_TRANSACTIONS
       0   No Brand Code                   2171
       1          KROGER                     65
       2  BEN AND JERRYS                     39
       3           BRAND                     24
       4        PRINGLES                     18
```

1.7 Entity Relationship Diagram

```
[ ]: def entity_relationship_diagram(df_1, df_2, df_3):

         erd = ERD()

         df_1 = df_1.rename(columns={'rewardsReceiptItemList.brandCode':␣
      ↪'rewardsReceiptItemList_brandCode'})

         t1 = erd.add_table(df_1, 'receipts_table', bg_color='pink')
         t2 = erd.add_table(df_2, 'user_table', bg_color='skyblue')
         t3 = erd.add_table(df_3, 'brand_table', bg_color='gold')

         erd.create_rel('receipts_table', 'brand_table',␣
      ↪left_on='rewardsReceiptItemList_brandCode', right_on='brandCode',␣
      ↪left_cardinality='*')
         erd.create_rel('receipts_table', 'user_table', left_on='userId',␣
      ↪right_on='_id', left_cardinality='*')

         erd.res = '\n'.join(erd.table_gen_code)

         print(erd.res)

     entity_relationship_diagram(RECEIPTSTBL, USERSTBL, BRANDTBL)
```



receipts_table
_id object
bonusPointsEarned float64
bonusPointsEarnedReason object
createDate int64
dateScanned int64
finishedDate float64
modifyDate int64
pointsAwardedDate float64
pointsEarned float64
purchaseDate datetime64[ns]
purchasedItemCount float64
rewardsReceiptStatus object
totalSpent float64
userId object
rewardsReceiptItemList.barcode object
rewardsReceiptItemList.description object
rewardsReceiptItemList.finalPrice object
rewardsReceiptItemList.itemPrice object
rewardsReceiptItemList.needsFetchReview object
rewardsReceiptItemList.partnerItemId object
rewardsReceiptItemList.preventTargetGapPoints object
rewardsReceiptItemList.quantityPurchased float64
rewardsReceiptItemList.userFlaggedBarcode object
rewardsReceiptItemList.userFlaggedNewItem object
rewardsReceiptItemList.userFlaggedPrice object
rewardsReceiptItemList.userFlaggedQuantity float64
rewardsReceiptItemList.needsFetchReviewReason object
rewardsReceiptItemList.pointsNotAwardedReason object
rewardsReceiptItemList.pointsPayerId object
rewardsReceiptItemList.rewardsGroup object
rewardsReceiptItemList.rewardsProductPartnerId object
rewardsReceiptItemList.userFlaggedDescription object
rewardsReceiptItemList.originalMetaBriteBarcode object
rewardsReceiptItemList.originalMetaBriteDescription object
rewardsReceiptItemList_brandCode object
rewardsReceiptItemList.competitorRewardsGroup object
rewardsReceiptItemList.discountedItemPrice object
rewardsReceiptItemList.originalReceiptItemText object
rewardsReceiptItemList.itemNumber object
rewardsReceiptItemList.originalMetaBriteQuantityPurchased float64
rewardsReceiptItemList.pointsEarned object
rewardsReceiptItemList.targetPrice object
rewardsReceiptItemList.competitiveProduct object
rewardsReceiptItemList.originalFinalPrice object
rewardsReceiptItemList.originalMetaBriteItemPrice object
rewardsReceiptItemList.deleted object
rewardsReceiptItemList.priceAfterCoupon object
rewardsReceiptItemList float64
rewardsReceiptItemList.metabriteCampaignId object

user_table
_id object
active bool
createdDate datetime64[ns]
lastLogin float64
role object
signUpSource object
state object

brand_table
_id object
barcode int64
category object
categoryCode object
name object
topBrand float64
brandCode object
cpg.$id.$oid object
cpg.$ref object

1.8 A Message to Stakeholders

To our stakeholders, please accept this follow up message on the questions posed about the previous discussed data sets:

- ***What are the top 5 brands by receipts scanned for most recent month?*** The most recent month in the receipts table is March 2021. There is only one brand code with scanned receipts in that month, that brand code being 'No Brand Code' with 2 receipts scanned.
- ***How does the ranking of the top 5 brands by receipts scanned for the recent month compare to the ranking for the previous month?*** "No Brand Code", being the only brand to have associated receipts scanned in the most recent month, saw a decrease of 166 scanned receipts between February 2021 and March 2021.
- ***When considering average spend from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?*** In the unique values for rewardsReceiptStatus, there is no value 'ACCEPTED'. So, 'REJECTED' would be greater by default, with an average value of $19.54
- ***When considering total number of items purchased from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?*** In the unique values for rewardsReceiptStatus, there is no value 'ACCEPTED'. So, 'REJECTED' would be greater by default, with a value of 167
- ***Which brand has the most spend among users who were created within the past 6 months?*** Here, we have used a time offset of 20 months in order to produce example results. The highest value belongs to "No Brand Code", followed by Kroger with a value of $222,538.59
- ***Which brand has the most transactions among users who were created within the past 6 months?*** Here, we have used a time offset of 20 months in order to produce example results. The highest value belongs to "No Brand Code", followed by KROGER with a transaction count of 65.

While there are several quality issues with the involved data sets, this reviewer observed most notably that the datetime formats between the three tables require conversion into a common unit for analysis. In the Receipts table, the 'Purchase Date' is in 13-digit Unix epoch time format, which is represented in milliseconds. While the timestamps in the Brands and Users tables are in 10-digit Unix epoch time format, which is represented in seconds. For the receipts table, it should also be noted that the brand code is not available for many for many transactions, making analysis by that metric challenging.

All my best,

Timothy Del Green 1-256-335-0378 tdgreen@outlook.com https://www.linkedin.com/in/timothy-del-green