Loading the Lookup Table

1. **Commands to load the relevant data in the Lookup Table**

"ranked_card_transactions_orc" table stores last 10 transactions for each card_id. Used ORC for better performance and "card_ucl_orc" table stores UCL values for each card_id Following commands were ran in Hive.

- Load data in **"ranked_card_transactions_orc**" table

```
INSERT OVERWRITE TABLE RANKED_CARD_TRANSACTIONS_ORC
SELECT B.CARD_ID, B.AMOUNT, B.POSTCODE, B.TRANSACTION_DT, B.RANK FROM
(SELECT A.CARD_ID, A.AMOUNT, A.POSTCODE, A.TRANSACTION_DT, RANK()
OVER(PARTITION BY A.CARD_IDORDER BY A.TRANSACTION_DT DESC, AMOUNT
DESC)
AS RANK FROM
(SELECT CARD_ID, AMOUNT, POSTCODE, TRANSACTION_DT FROM
CARD_TRANSACTIONS_HBASE WHERESTATUS = 'GENUINE') A ) B WHERE B.RANK <=
10
```

- Load data in "**card_ucl_orc**" table. In innermost query, select card_id, average of amount and standard deviation of amount from card_transactions_orc. In outermost query, select card_id and compute UCL using average and standard deviation with formula (avg + (3 * stddev)). Insert all this data in card_ucl_orc

```
INSERT OVERWRITE TABLE CARD_UCL_ORC
SELECT A.CARD_ID, (A.AVERAGE + (3 * A.STANDARD_DEVIATION)) AS UCL FROM (
SELECT CARD_ID, AVG(AMOUNT) AS AVERAGE, STDDEV(AMOUNT) AS
STANDARD_DEVIATION FROM RANKED_CARD_TRANSACTIONS_ORC
GROUP BY CARD_ID) A;
```

- Load data in **lookup_data_hbase table.**

```
INSERT OVERWRITE TABLE LOOKUP_DATA_HBASE
SELECT RCTO.CARD_ID, CUO.UCL, CMS.SCORE, RCTO.POSTCODE,
RCTO.TRANSACTION_DT FROM RANKED_CARD_TRANSACTIONS_ORC RCTO
JOIN CARD_UCL_ORC CUO ON CUO.CARD_ID = RCTO.CARD_ID JOIN (SELECT
DISTINCT
CARD.CARD_ID, SCORE.SCORE FROM CARD_MEMBER_ORC CARD
JOIN MEMBER_SCORE_ORC SCORE ON CARD.MEMBER_ID = SCORE.MEMBER_ID) AS
CMS ON RCTO.CARD_ID = CMS.CARD_ID WHERE RCTO.RANK = 1
```

2. <Command to see the table created and its content>

- Verify data in **"lookup_data_hbase"** table.
  select * from lookup_data_hbase limit 10;
- Verify count in "**lookup_data_hive**" table.
  count 'lookup_data_hive'
- Verify data in "**lookup_data_hive**" table.
  scan 'lookup_data_hive'

2. Screenshot of the created table

- Load data in **"ranked_card_transactions_orc"** table

```
hive> INSERT OVERWRITE TABLE RANKED_CARD_TRANSACTIONS_ORC SELECT B.CARD_ID, B.AMOUNT, B.POSTCODE, B.TRANSACTION_DT, B.RANK FROM (SELECT A.CARD_ID, A.AMOUNT,
A.POSTCODE, A.TRANSACTION_DT, RANK() OVER(PARTITION BY A.CARD_ID ORDER BY A.TRANSACTION_DT DESC, AMOUNT DESC) AS RANK FROM (SELECT CARD_ID, AMOUNT, POSTCODE,
 TRANSACTION_DT FROM CARD_TRANSACTIONS_HBASE WHERE STATUS = 'GENUINE') A ) B WHERE B.RANK <= 10;
Query ID = root_20240716061632_d02d31ac-9339-428d-93cb-ba93e668cfc7
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1721107016086_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 11.36 s
--------------------------------------------------------------------------------
Loading data to table capstone_project.ranked_card_transactions_orc
OK
Time taken: 25.016 seconds
hive>
```

- Load data in "**card_ucl_orc**" table

```
hive> INSERT OVERWRITE TABLE CARD_UCL_ORC
    > SELECT A.CARD_ID, (A.AVERAGE + (3 * A.STANDARD_DEVIATION)) AS UCL FROM (
    > SELECT CARD_ID, AVG(AMOUNT) AS AVERAGE, STDDEV(AMOUNT) AS STANDARD_DEVIATION
    > FROM RANKED_CARD_TRANSACTIONS_ORC
    > GROUP BY CARD_ID) A;
Query ID = root_20240716061734_bc794501-7d10-4d61-8fb7-30bf9af0af9a
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1721107016086_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 6.55 s
--------------------------------------------------------------------------------
Loading data to table capstone_project.card_ucl_orc
OK
Time taken: 8.562 seconds
hive>
```

- Load data in “**lookup_data_hbase**” table.

```
hive> INSERT OVERWRITE TABLE LOOKUP_DATA_HBASE SELECT RCTO.CARD_ID, CUO.UCL, CMS.SCORE, RCTO.POSTCODE, RCTO.TRANSACTION_DT FROM RANKED_CARD_TRANSACTIONS_ORC
RCTO JOIN CARD_UCL_ORC CUO ON CUO.CARD_ID = RCTO.CARD_ID JOIN ( SELECT DISTINCT CARD.CARD_ID, SCORE.SCORE FROM CARD_MEMBER_ORC CARD JOIN MEMBER_SCORE_ORC SCO
RE ON CARD.MEMBER_ID = SCORE.MEMBER_ID) AS CMS ON RCTO.CARD_ID = CMS.CARD_ID WHERE RCTO.RANK = 1;
No Stats for capstone_project@ranked_card_transactions_orc, Columns: postcode, rank, transaction_dt, card_id
No Stats for capstone_project@card_ucl_orc, Columns: card_id, ucl
No Stats for capstone_project@card_member_orc, Columns: member_id, card_id
No Stats for capstone_project@member_score_orc, Columns: member_id, score
Query ID = root_20240716061946_b94fa708-e503-43c5-a074-bf1232413eca
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1721107016086_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Map 2 .......... container     SUCCEEDED      1          1        0        0       0       0
Map 3 .......... container     SUCCEEDED      1          1        0        0       0       0
Map 5 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 4 ...... container     SUCCEEDED      2          2        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 05/05  [==========================>>] 100%  ELAPSED TIME: 18.01 s
--------------------------------------------------------------------------------
OK
Time taken: 22.188 seconds
hive>
```

- Verify some data in “**lookup_data_hbase**” table.

```
hive> select * from lookup_data_hbase limit 10;
OK
340028465709212 1.6331555548882348E7    233     24658   2018-01-02 03:25:35
340054675199675 1.4156079786189131E7    631     50140   2018-01-15 19:43:23
340082915339645 1.5285685330791473E7    407     17844   2018-01-26 19:03:47
340134186926007 1.5239767522438556E7    614     67576   2018-01-18 23:12:50
340265728490548 1.608491671255562E7     202     72435   2018-01-21 02:07:35
340268219434811 1.2507323937605347E7    415     62513   2018-01-16 04:30:05
340379737226464 1.4198310998368107E7    229     26656   2018-01-27 00:19:47
340383645652108 1.4091750460468251E7    645     34734   2018-01-29 01:29:12
340803866934451 1.0843341196185412E7    502     87525   2018-01-31 04:23:57
340889618969736 1.3217942365515321E7    330     61341   2018-01-31 21:57:18
Time taken: 0.304 seconds, Fetched: 10 row(s)
hive>
```

- Verify count in “**lookup_data_hive**” table.

```
hbase(main):001:0> count 'lookup_data_hive'
999 row(s) in 0.4200 seconds

=> 999
hbase(main):002:0>
```

Total number for record is 999 which is matching with given requirement.

- Verify data in “lookup_data_hive” table.

```
 6595928469079750                    column=lookup_card_family:score, timestamp=1721110808662, value=412
 6595928469079750                    column=lookup_card_family:ucl, timestamp=1721110808662, value=1.142797041440079E7
 6595928469079750                    column=lookup_transaction_family:postcode, timestamp=1721110808662, value=98349
 6595928469079750                    column=lookup_transaction_family:transaction_dt, timestamp=1721110808662, value=2018-01-24 12:38:22
 6597703848279563                    column=lookup_card_family:score, timestamp=1721110808662, value=218
 6597703848279563                    column=lookup_card_family:ucl, timestamp=1721110808662, value=1.4718634149498457E7
 6597703848279563                    column=lookup_transaction_family:postcode, timestamp=1721110808662, value=95699
 6597703848279563                    column=lookup_transaction_family:transaction_dt, timestamp=1721110808662, value=2018-01-27 10:51:49
 6598830758632447                    column=lookup_card_family:score, timestamp=1721110808662, value=293
 6598830758632447                    column=lookup_card_family:ucl, timestamp=1721110808662, value=1.2227949982601807E7
 6598830758632447                    column=lookup_transaction_family:postcode, timestamp=1721110808662, value=19421
 6598830758632447                    column=lookup_transaction_family:transaction_dt, timestamp=1721110808662, value=2018-01-30 00:18:34
 6599900931314251                    column=lookup_card_family:score, timestamp=1721110808662, value=297
 6599900931314251                    column=lookup_card_family:ucl, timestamp=1721110808662, value=1.2121408572464656E7
 6599900931314251                    column=lookup_transaction_family:postcode, timestamp=1721110808662, value=97423
 6599900931314251                    column=lookup_transaction_family:transaction_dt, timestamp=1721110808662, value=2018-01-31 11:25:16
999 row(s) in 6.2530 seconds

hbase(main):003:0>
```

Total number for record is 999 which is matching with given requirement