CAPSTONE Project

1) creating table card_transaction in hive

```
CREATE TABLE IF NOT EXISTS card_transactions
(card_id BIGINT ,
member_id BIGINT,
amount BIGINT,
postcode BIGINT,
pos_id BIGINT,
transaction_dt string,
status string)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ',' STORED AS TEXTFILE
tblproperties("skip.header.line.count"="1");
LOAD DATA INPATH '/user/ec2-user/input/card_transactions.csv'
overwrite INTO TABLE card_transactions;
```

2) adding a unique row Id to uniquely identify a record

```
CREATE TABLE card_transactions_staging
AS
SELECT
  regexp_replace(reflect('java.util.UUID','randomUUID'), '-', '') as rowid,
  *
FROM card_transactions
```

3) creating an intermediary hive table to load data in NoSQL table(Hbase)

```
create table IF not exists card_transaction_hive
(row_key string,card_id BIGINT ,member_id BIGINT,amount BIGINT,postcode
BIGINT,pos_id BIGINT,transaction_dt string,status string)
STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
WITH SERDEPROPERTIES ("hbase.columns.mapping" =
":key,cf:card_id,cf:member_id,cf:amount,cf:postcode,cf:pos_id,cf:transaction_dt,cf:status
")
TBLPROPERTIES ("hbase.table.name" = "card_transaction_hbase");
```

4) loading data into Hbase table via hive table

```
INSERT OVERWRITE TABLE card_transaction_hive SELECT
rowid,card_id ,member_id,amount,postcode,pos_id,string(unix_timestamp(transaction_dt,
'dd-MM-yyyy HH:mm:ss')),status
FROM card_transactions_staging;
```

5)**Importing AWS RDS data into HDFS**

```
sqoop import --connect jdbc:mysql://upgradawsrds.cpclxrkdvwmz.us-
east-1.rds.amazonaws.com:3306/cred_financials_data --username upgraduser --
password upgraduser --table card_member --warehouse-dir input/tables/navjot/
```

```
CREATE TABLE IF NOT EXISTS card_member
(card_id BIGINT ,
member_id BIGINT,
member_joining_dt string,
card_purchase_dt string,
country string,
city string)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ',' STORED AS TEXTFILE
tblproperties("skip.header.line.count"="1");
```

6) Loading the data into hive table from hdfs

```
LOAD DATA INPATH '/user/root/input/tables/navjot/card_member'
overwrite INTO TABLE card_member;
```

```
sqoop import --connect jdbc:mysql://upgradawsrds.cpclxrkdvwmz.us-
east-1.rds.amazonaws.com:3306/cred_financials_data --username upgraduser --
password upgraduser --table member_score --warehouse-dir input/tables/navjot/
```

7_ **Creating table in hive table**

```
CREATE TABLE IF NOT EXISTS member_score
(member_id BIGINT ,
score BIGINT)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ',' STORED AS TEXTFILE
tblproperties("skip.header.line.count"="1");
```

8 loading data in table in hive

```
LOAD DATA INPATH '/user/root/input/tables/navjot/member_score'
overwrite INTO TABLE member_score;
```

9)UCL table

```
CREATE TABLE IF NOT EXISTS User_UCL AS
SELECT b.card_id, (avg(b.amount) + 3*(stddev(b.amount)))
FROM
(
  SELECT card_id, amount, transaction_dt,
       row_number() over (partition by card_id order by transaction_dt desc) r
  FROM card_transactions where status like 'GENUINE'
) b
WHERE r <= 10
GROUP BY b.card_id;
```

10) user look up table

```
CREATE   TABLE   IF   NOT   EXISTS  User_lookup AS

SELECT card_transactions.card_id,
card_transactions.postcode,card_transactions.transaction_dt,user_ucl.`_c1`
FROM card_transactions
  INNER JOIN (SELECT card_id, MAX(transaction_dt) As MaxDate FROM
card_transactions  GROUP BY card_id) B
    ON card_transactions.card_id = B.card_id AND card_transactions.transaction_dt =
B.MaxDate
Join user_ucl
ON card_transactions.card_id = user_ucl.card_id
```