HIVE CASE STUDY

PREPARED BY ASHAR HABIB

UPGRAD IIITB LJMU C3 STUDENT.

Problem Statement :

With online sales gaining popularity, tech companies are exploring ways to improve their sales by analyzing customer behaviors and gaining insights about product trends. Furthermore, the websites make it easier for customers to find the products they require without much scavenging. Needless to say, the role of big data analysts is among the most sought-after job profiles of this decade. Therefore, as part of this assignment, we will be challenging you, as a big data analyst, to extract data and gather insights from a real-life data set of an e-commerce company.

For this assignment, I will be working with a public clickstream dataset of a cosmetics store. Using this dataset, our job is to extract valuable insights which generally data engineers come up within an e-retail company.

we will find the data in the link given below.

https://e-commerce-events-ml.s3.amazonaws.com/2019-Oct.csv
https://e-commerce-events-ml.s3.amazonaws.com/2019-Nov.csv

Case study Objectives to provide answers to the questions given below:

1. Find the total revenue generated due to purchases made in October.
2. Write a query to yield the total sum of purchases per month in a single output.
3. Write a query to find the change in revenue generated due to purchases from October to November.
4. Find distinct categories of products. Categories with null category code can be ignored.
5. Find the total number of products available under each category.
6. Which brand had the maximum sales in October and November combined?
7. Which brands increased their sales from October to November?
8. Your company wants to reward the top 10 users of its website with a Golden Customer plan. Write a query to generate a list of top 10 users who spend the most.

STEP 1: AFTER LAUNCHING AN EMR CLUSTER. MOVE THE DATA FROM S3 BUCKET INTO HDFS.

```
[hadoop@ip-10-0-5-147 ~]$ hadoop fs -mkdir /tmp/meta_data
[hadoop@ip-10-0-5-147 ~]$ aws s3 cp s3://e-commerce-events-ml/2019-Oct.csv .
download: s3://e-commerce-events-ml/2019-Oct.csv to ./2019-Oct.csv
[hadoop@ip-10-0-5-147 ~]$ hadoop fs -put 2019-Oct.csv /tmp/meta_data
[hadoop@ip-10-0-5-147 ~]$ aws s3 cp s3://e-commerce-events-ml/2019-Nov.csv .
download: s3://e-commerce-events-ml/2019-Nov.csv to ./2019-Nov.csv
[hadoop@ip-10-0-5-147 ~]$ hadoop fs -put 2019-Nov.csv /tmp/meta_data
[hadoop@ip-10-0-5-147 ~]$ ls -list
total 1004292
38594 533052 -rw-rw-r-- 1 hadoop hadoop 545839412 Mar 17  2020 2019-Nov.csv
38593 471240 -rw-rw-r-- 1 hadoop hadoop 482542278 Mar 17  2020 2019-Oct.csv
```

STEP 2 : HERE WE ARE USING CSV SERDE WITH DEFAULT PROPERTIES VALUES FOR LOADING THE DATASET INTO HIVE TABLE, CREATE THE DATABASE:

```
[hadoop@ip-10-0-5-147 ~]$ hive
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> create database if not exists cosmetics_db;
OK
Time taken: 0.621 seconds
USE cosmetics_db;
OK
```

STEP 3 : AFTER CREATE AND USE THE DATABASE DATA BASE AND SCHEMA:

```
hive> describe database extended cosmetics_db;
OK
cosmetics_db            hdfs://ip-10-0-5-147.ec2.internal:8020/user/hive/warehouse/cosmetics_db.db      hadoop
USER
Time taken: 0.279 seconds, Fetched: 1 row(s)
hive> show databases;
OK
cosmetics_db
default
Time taken: 0.039 seconds, Fetched: 2 row(s)
```

```
 hive> describe schema cosmetics_db;
 OK
 cosmetics_db            hdfs://ip-10-0-5-147.ec2.internal:8020/user/hive/warehouse/cosmetics_db.db      hadoop  USER
 Time taken: 0.035 seconds, Fetched: 1 row(s)
```

STEP 4 : CREATE THE EXTERNAL TABLE AND CHECK THE STRUCTURE OF THE TABLE:

```
hive> create external table if not exists test_data (event_time timestamp, event_type string,product_id string,
 category_id string, category_code string, brand string, price float,user_id bigint, user_session string)
 ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH SERDEPROPERTIES('separatorChar' = ',') STORED AS
TEXTFILE LOCATION '/tmp/meta_data/' TBLPROPERTIES('skip.header.line.count' = '1');
 OK
Time taken: 0.146 seconds
hive> desc test_data;
OK
event_time              string                  from deserializer
event_type              string                  from deserializer
product_id              string                  from deserializer
category_id             string                  from deserializer
category_code           string                  from deserializer
brand                   string                  from deserializer
price                   string                  from deserializer
user_id                 string                  from deserializer
user_session            string                  from deserializer
Time taken: 0.083 seconds, Fetched: 9 row(s)
```

```
hive> describe formatted test_data;
OK
# col_name              data_type               comment

event_time              string                  from deserializer
event_type              string                  from deserializer
product_id              string                  from deserializer
category_id             string                  from deserializer
category_code           string                  from deserializer
brand                   string                  from deserializer
price                   string                  from deserializer
user_id                 string                  from deserializer
user_session            string                  from deserializer

# Detailed Table Information
Database:               default
Owner:                  hadoop
CreateTime:             Sat Oct 01 13:25:23 UTC 2022
LastAccessTime:         UNKNOWN
Retention:              0
Location:               hdfs://ip-10-0-5-147.ec2.internal:8020/tmp/meta_data
Table Type:             EXTERNAL_TABLE
Table Parameters:
        EXTERNAL                TRUE
        numFiles                2
        skip.header.line.count  1
        totalSize               1028381690
        transient_lastDdlTime   1664630723
```

```
# Storage Information
SerDe Library:          org.apache.hadoop.hive.serde2.OpenCSVSerde
InputFormat:            org.apache.hadoop.mapred.TextInputFormat
OutputFormat:
org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        separatorChar           ,
        serialization.format    1
Time taken: 0.107 seconds, Fetched: 38 row(s)
hive> create external table if not exists store_data (event_time
timestamp, event_type string,product_id string,
 category_id string, category_code string, brand string, price
float,user_id bigint, user_session string);
OK
Time taken: 0.061 seconds
```

STEP 5: WE FIND THE DATA TYPES ALL ARE IN STRING WE NEED TO CAST THEM TO THE DESIRED ONE:

```
hive> create external table if not exists store_data (event_time timestamp, event_type
string,product_id string,
 category_id string, category_code string, brand string, price float,user_id bigint,
user_session string);
OK
Time taken: 0.061 seconds
hive> insert into store_data select cast (from_unixtime(unix_timestamp(event_time,'yyyy-
MM-dd HH:mm:ss Z'),
'yyyy-MM-dd HH:mm:ss')as timestamp) as event_time, event_type,product_id, category_id,
category_code,brand,
cast(price as float) as price, cast(user_id as bigint) as user_id, user_session from test_data;

Query ID = hadoop_20221001132848_cb0d09b9-b2d5-4725-861f-c04f5f4dc71b
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_1664627482233_0002)
```

```
Map 1: 0/2
Map 1: 0/2
Map 1: 0(+1)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
```

```
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 0(+2)/2
Map 1: 1(+1)/2
Map 1: 1(+1)/2
Map 1: 1(+1)/2
Map 1: 2/2
Loading data to table default.store_data
OK
Time taken: 161.567 seconds
```

```
hive> describe store_data;
OK
event_time              timestamp
event_type              string
product_id              string
category_id             string
category_code           string
brand                   string
price                   float
user_id                 bigint
user_session            string
Time taken: 0.076 seconds, Fetched: 9 row(s)

hive> show tables in cosmetics_db;
OK
store_data
test_data
Time taken: 0.022 seconds, Fetched: 2 row(s)


hive> set hive.cli.print.header=true;
hive> select event_type, count(event_type) as count from store_data group by event_type;
Query ID = hadoop_20221001133603_02ce2c4e-5b94-4343-8ede-3a53a176d9bb
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0002)
```

```
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+2)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+2)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 5(+2)/7  Reducer 2: 0/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 1(+2)/4
Map 1: 7/7      Reducer 2: 1(+3)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 4/4
OK
event_type      count
view    3938296
purchase        568041
cart    2544192
remove_from_cart        1687591
Time taken: 31.057 seconds, Fetched: 4 row(s)
```

Note: here we can see that view count is more then purchases.

QUERY # 1 . Find the total revenue generated due to purchases made in October.

```
hive> select sum(price) as oct_revenue from store_data where month(event_time)='10' and
event_type='purchase';
Query ID = hadoop_20221001133924_a248563e-2083-4b87-b908-8c3a1cedc0f3
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0002)

Map 1: 0/7      Reducer 2: 0/1
Map 1: 0/7      Reducer 2: 0/1
Map 1: 0/7      Reducer 2: 0/1
Map 1: 0(+1)/7  Reducer 2: 0/1
Map 1: 0(+2)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 1(+3)/7  Reducer 2: 0/1
Map 1: 2(+3)/7  Reducer 2: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/1
Map 1: 4(+3)/7  Reducer 2: 0/1
Map 1: 5(+2)/7  Reducer 2: 0(+1)/1
Map 1: 6(+1)/7  Reducer 2: 0(+1)/1
Map 1: 7/7      Reducer 2: 1/1
OK
oct_revenue
1211538.4295325726
Time taken: 39.073 seconds, Fetched: 1 row(s)
```

Note:
One of the optimization technique is partition, to increase the performance apply partition here and compare the execution time:

Static Partition:

```
hive> create external table if not exists purchase_data(event_time timestamp, product_id
string,category_id string,
 category_code string, brand string, price float, user_id bigint,user_session string) partitioned
by (event_type string)
 row format delimited fields terminated by "," lines terminated by "\n" stored as textfile;

OK
Time taken: 0.08 seconds

hive> insert into table purchase_data partition(event_type = "purchase")select event_time,
product_id,
 category_id, category_code, brand, price, user_id, user_session from store_data where
event_type = 'purchase';
Query ID = hadoop_20221001135206_1c1ccd6f-453d-48e6-9870-0f10e5e9eb48
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_1664627482233_0003)
```

```
Map 1: 0/7
Map 1: 0/7
Map 1: 0/7
Map 1: 0(+1)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 1(+3)/7
Map 1: 2(+2)/7
Map 1: 2(+3)/7
Map 1: 3(+2)/7
Map 1: 3(+3)/7
Map 1: 3(+3)/7
Map 1: 4(+3)/7
Map 1: 5(+2)/7
Map 1: 6(+1)/7
Map 1: 7/7
Loading data to table default.purchase_data partition (event_type=purchase)
OK
event_time     product_id     category_id     category_code   brand   price   user_id   user_session
Time taken: 47.014 seconds

hive> show partitions purchase_data;
OK
partition
event_type=purchase
Time taken: 0.079 seconds, Fetched: 1 row(s)
```

```
hive> show tables;
OK
tab_name
purchase_data
store_data
test_data
Time taken: 0.031 seconds, Fetched: 3 row(s)
hive> select * from purchase_data limit 5;
OK
purchase_data.event_time        purchase_data.product_id        purchase_data.category_id
purchase_data.category_code     purchase_data.brand     purchase_data.price     purchase_data.user_id
purchase_data.user_session      purchase_data.event_type
2019-11-01 00:01:57     5839412 1487580006551913373             lovely  3.16   460304619
9f777569-bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:01:57     5823969 1487580005268456287             uno     17.46  460304619
9f777569-bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:01:57     5810480 1487580011283087468                     22.54  460304619       9f777569-
bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:04:33     24380   1487580012994363565             depilflax      5.24     564451209
861ab2f1-b2e5-886f-a93b-5b067eff081f    purchase
2019-11-01 00:04:33     26765   1487580013522845895             ardell  7.16   564451209       861ab2f1-
b2e5-886f-a93b-5b067eff081f    purchase
Time taken: 0.24 seconds, Fetched: 5 row(s)
```

```
hive> select sum(price) as oct_revenue from purchase_data where month(event_time)="10";
Query ID = hadoop_20221001135440_0fd66102-91bb-4942-aa5a-d2d4e85b0328
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0003)

Map 1: 0/3      Reducer 2: 0/1
Map 1: 0/3      Reducer 2: 0/1
Map 1: 0/3      Reducer 2: 0/1
Map 1: 0(+1)/3  Reducer 2: 0/1
Map 1: 0(+2)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 1(+2)/3  Reducer 2: 0/1
Map 1: 1(+2)/3  Reducer 2: 0(+1)/1
Map 1: 3/3      Reducer 2: 0(+1)/1
Map 1: 3/3      Reducer 2: 0/1
Map 1: 3/3      Reducer 2: 1/1
OK
oct_revenue
1211538.4295325726
Time taken: 24.427 seconds, Fetched: 1 row(s)
```

CREATE DYNAMIC PARTITION:

```
set hive.exec.dynamic.partition=true;
set hive.exec.dynamic.partition.mode=nonstrict;
```

```
hive> create external table if not exists mnth_dyn_data (event_type string,product_id string,
category_id string,
 category_code string, brand string, price float,  user_id bigint, user_session string) partitioned
by (event_time string)
 row format delimited fields terminated by "," lines terminated by "\n" stored as textfile;

OK
Time taken: 0.076 seconds
hive> insert into mnth_dyn_data partition(event_time) select event_type, product_id,
category_id, category_code,
 brand, price, user_id, user_session, substr(event_time, 6,2) from store_data;

Query ID = hadoop_20221001135652_39c8f743-1902-41cf-af91-c991e96ebe37
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0003)
```

```
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+1)/7  Reducer 2: 0/4
Map 1: 0(+2)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 5(+2)/7  Reducer 2: 0(+1)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 1(+3)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 4/4
```

```
Loading data to table default.mnth_dyn_data partition (event_time=null)


        Time taken to load dynamic partitions: 0.243 seconds
        Time taken for adding to write entity : 0.001 seconds
OK
event_type      product_id      category_id     category_code   brand   price   user_id user_session
_c8
Time taken: 123.374 seconds
```

```
hive> show tables;
OK
tab_name
mnth_dyn_data
purchase_data
store_data
test_data
Time taken: 0.033 seconds, Fetched: 4 row(s)
hive> show partitions mnth_dyn_data;
OK
partition
event_time=10
event_time=11
Time taken: 0.059 seconds, Fetched: 2 row(s)
hive> select sum(price) as oct_revenue from mnth_dyn_data where event_time="10" and
event_type ="purchase";
Query ID = hadoop_20221001140013_60c62702-6d1d-472b-93db-7e908ec0407e
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0003)
```

```
Map 1: 0/5      Reducer 2: 0/1
Map 1: 0/5      Reducer 2: 0/1
Map 1: 0/5      Reducer 2: 0/1
Map 1: 0(+2)/5  Reducer 2: 0/1
Map 1: 0(+3)/5  Reducer 2: 0/1
Map 1: 0(+3)/5  Reducer 2: 0/1
Map 1: 0(+3)/5  Reducer 2: 0/1
Map 1: 0(+3)/5  Reducer 2: 0/1
Map 1: 1(+2)/5  Reducer 2: 0/1
Map 1: 2(+1)/5  Reducer 2: 0/1
Map 1: 3(+1)/5  Reducer 2: 0/1
Map 1: 3(+2)/5  Reducer 2: 0(+1)/1
Map 1: 5/5      Reducer 2: 0(+1)/1
Map 1: 5/5      Reducer 2: 1/1
OK
oct_revenue
1211538.4295325726
Time taken: 24.306 seconds, Fetched: 1 row(s)
```

Note:

After Dynamic partition the execution time reduced around 50%.

Now create Bucketing :

```
set hive.enforce.bucketing = true;
set hive.exec.max.dynamic.partitions.pernode=1000;
```

```
hive> create external table if not exists test_bucket_data (event_type string, product_id string,
category_id string, category_code string, brand string, price float, user_id bigint, user_session
string)partitioned by (event_time string) clustered by (event_type) into 3 buckets row format
delimited fields terminated by "," lines terminated by "\n" stored as textfile;
OK
Time taken: 0.853 seconds
hive> show tables;
OK
mnth_dyn_data
purchase_data
store_data
test_bucket_data
test_data
Time taken: 0.089 seconds, Fetched: 5 row(s)
```

```
FAILED: SemanticException [Error 10096]: Dynamic partition strict mode
requires at least one static partition column.To turn this off set
hive.exec.dynamic.partition.mode=nonstrict
```

```
hive> set hive.exec.dynamic.partition.mode=nonstrict;
hive> insert into test_bucket_data partition (event_time) select event_type, product_id,
category_id, category_code, brand, price, user_id, user_session, substr(event_time, 6,2) from
store_data;
Query ID = hadoop_20221001180100_d1d4bb0d-5d9f-402f-9627-45e4d00a3e69
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0009)
```

```
Map 1: -/-      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+1)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 5(+2)/7  Reducer 2: 0(+1)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 1(+2)/4
Map 1: 7/7      Reducer 2: 1(+3)/4
Map 1: 7/7      Reducer 2: 1(+3)/4
```

```
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 4/4
Loading data to table default.test_bucket_data partition
(event_time=null)
```

```
 Time taken to load dynamic partitions: 0.434 seconds
         Time taken for adding to write entity : 0.003 seconds
OK
Time taken: 161.856 seconds
hive> show tables;
OK
mnth_dyn_data
purchase_data
store_data
test_bucket_data
test_data
Time taken: 0.029 seconds, Fetched: 5 row(s)
```

```
hive> select sum(price) from test_bucket_data where event_type='purchase' and
event_time=10;
Query ID = hadoop_20221001180920_02321d2a-cf32-46ae-b5bc-e212a6a98130
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_1664627482233_0010)
```

```
Map 1: -/-      Reducer 2: 0/1
Map 1: 0/6      Reducer 2: 0/1
Map 1: 0/6      Reducer 2: 0/1
Map 1: 0/6      Reducer 2: 0/1
Map 1: 0(+1)/6  Reducer 2: 0/1
Map 1: 0(+2)/6  Reducer 2: 0/1
Map 1: 0(+3)/6  Reducer 2: 0/1
Map 1: 0(+3)/6  Reducer 2: 0/1
Map 1: 0(+3)/6  Reducer 2: 0/1
Map 1: 0(+3)/6  Reducer 2: 0/1
Map 1: 1(+2)/6  Reducer 2: 0/1
Map 1: 2(+2)/6  Reducer 2: 0/1
Map 1: 3(+1)/6  Reducer 2: 0/1
Map 1: 3(+2)/6  Reducer 2: 0/1
Map 1: 3(+3)/6  Reducer 2: 0/1
Map 1: 4(+2)/6  Reducer 2: 0(+1)/1
Map 1: 5(+1)/6  Reducer 2: 0(+1)/1
Map 1: 6/6      Reducer 2: 0(+1)/1
Map 1: 6/6      Reducer 2: 1/1
OK
1211538.4295325726
Time taken: 37.209 seconds, Fetched: 1 row(s)
```

Note:

we can create direct static partition table from 2019_Oct.csv for this Query:

```
hive> create external table if not exists oct_data_1 (event_time timestamp, product_id string,
category_id string, category_code string, brand string, price float, user_id bigint, user_session
string)partitioned by (event_type string) row format delimited fields terminated by "," lines terminated
by "\n" stored as textfile;
OK
Time taken: 0.091 seconds
```

```
hive> insert into table oct_data_1 partition(event_type = 'purchase') select event_time,
product_id, category_id, category_code, brand, price, user_id, user_session from store_data
where event_type = 'purchase';
Query ID = hadoop_20221001182204_da52d93c-e877-4efa-a748-65b941178f3a
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_1664627482233_0011)
```

```
Map 1: 0/7
Map 1: 0/7
Map 1: 0/7
Map 1: 0(+1)/7
Map 1: 0(+2)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 0(+3)/7
Map 1: 1(+3)/7
Map 1: 2(+3)/7
Map 1: 3(+3)/7
Map 1: 3(+3)/7
Map 1: 4(+3)/7
Map 1: 5(+2)/7
Map 1: 6(+1)/7
Map 1: 7/7
Loading data to table default.oct_data_1 partition
(event_type=purchase)
OK
Time taken: 46.99 seconds
```

```
hive> select * from oct_data_1 limit 5;
OK
2019-11-01 00:01:57     5839412 1487580006551913373             lovely  3.16    460304619
9f777569-bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:01:57     5823969 1487580005268456287             uno     17.46   460304619
9f777569-bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:01:57     5810480 1487580011283087468                     22.54   460304619
9f777569-bdf3-47e5-a3d4-dfc26beb29cb    purchase
2019-11-01 00:04:33     24380   1487580012994363565             depilflax       5.24
564451209       861ab2f1-b2e5-886f-a93b-5b067eff081f    purchase
2019-11-01 00:04:33     26765   1487580013522845895             ardell  7.16    564451209
861ab2f1-b2e5-886f-a93b-5b067eff081f    purchase
Time taken: 0.377 seconds, Fetched: 5 row(s)
```

```
hive> select sum(price) as oct_revenue from oct_data_1 where month(event_time)=10;
Query ID = hadoop_20221001182508_343b545e-cc80-43f0-96f2-93ffa4df9028
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0011)

Map 1: 0/3      Reducer 2: 0/1
Map 1: 0/3      Reducer 2: 0/1
Map 1: 0/3      Reducer 2: 0/1
Map 1: 0(+1)/3  Reducer 2: 0/1
Map 1: 0(+2)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1
Map 1: 1(+2)/3  Reducer 2: 0(+1)/1
Map 1: 2(+1)/3  Reducer 2: 0(+1)/1
Map 1: 3/3      Reducer 2: 0(+1)/1
Map 1: 3/3      Reducer 2: 1/1
OK
1211538.4295325726
Time taken: 23.363 seconds, Fetched: 1 row(s)
```

NOTE: HERE WE CAN SEE THAT THE EXECUTION TIME CHANGE FORM 46.9 TO 23.3 SECONDS

QUERY 2: Write a querry to yield the total sum of purchases per month in a single output?

```
hive> select month(event_time) as month, sum(price) as revenue from store_data where event_type='purchase' group by month(event_time);
Query ID = hadoop_20221001182610_0ed970bd-a808-4173-89a3-a9337e3567c2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0011)

Map 1: 0/7      Reducer 2: 0/2
Map 1: 0/7      Reducer 2: 0/2
Map 1: 0/7      Reducer 2: 0/2
Map 1: 0(+2)/7  Reducer 2: 0/2
Map 1: 0(+3)/7  Reducer 2: 0/2
Map 1: 0(+3)/7  Reducer 2: 0/2
Map 1: 0(+3)/7  Reducer 2: 0/2
Map 1: 0(+3)/7  Reducer 2: 0/2
Map 1: 0(+3)/7  Reducer 2: 0/2
Map 1: 1(+3)/7  Reducer 2: 0/2
Map 1: 3(+3)/7  Reducer 2: 0/2
Map 1: 3(+3)/7  Reducer 2: 0/2
Map 1: 4(+2)/7  Reducer 2: 0/2
Map 1: 4(+3)/7  Reducer 2: 0/2
Map 1: 5(+2)/7  Reducer 2: 0/2
Map 1: 6(+1)/7  Reducer 2: 0(+1)/2
Map 1: 6(+1)/7  Reducer 2: 0(+2)/2
Map 1: 7/7      Reducer 2: 0(+2)/2
Map 1: 7/7      Reducer 2: 1(+1)/2
Map 1: 7/7      Reducer 2: 2/2
OK
10      1211538.4295325726
11      1531016.8991247676
Time taken: 30.455 seconds, Fetched: 2 row(s)
```

QUERY 3: write a querry to find the change in revenue generated due to purchases from October to November.

```
hive> with change_in_revenue as ( select sum( case  when month(event_time)="10" then price
else 0 end) as oct_rev, sum(case when month(event_time)="11" then price else 0 end) as
nov_rev  from store_data where event_type = 'purchase')select abs(oct_rev - nov_rev) as
change_in_rev from change_in_revenue;
Query ID = hadoop_20221001183813_e720012e-d65f-4fa1-8db1-273c36f967cd
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_1664627482233_0012)
```

```
Map 1: -/-      Reducer 2: 0/1
Map 1: 0/7      Reducer 2: 0/1
Map 1: 0/7      Reducer 2: 0/1
Map 1: 0/7      Reducer 2: 0/1
Map 1: 0(+1)/7  Reducer 2: 0/1
Map 1: 0(+2)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/1
Map 1: 1(+2)/7  Reducer 2: 0/1
Map 1: 1(+3)/7  Reducer 2: 0/1
Map 1: 2(+3)/7  Reducer 2: 0/1
Map 1: 3(+2)/7  Reducer 2: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/1
Map 1: 4(+3)/7  Reducer 2: 0/1
Map 1: 5(+2)/7  Reducer 2: 0/1
Map 1: 5(+2)/7  Reducer 2: 0(+1)/1
Map 1: 6(+1)/7  Reducer 2: 0(+1)/1
Map 1: 7/7      Reducer 2: 0(+1)/1
Map 1: 7/7      Reducer 2: 1/1
OK
319478.469592195
Time taken: 40.893 seconds, Fetched: 1
row(s)
```

QUERY 4: Find distinct categories of products. Categories with null category code can be ignored?

```
hive> select distinct category_code as product_category from store_data where category_code is not null ;
Query ID = hadoop_20221001183942_ee0dabc0-65de-4a68-a2e0-2e0aaab436b2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0012)

Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+1)/7  Reducer 2: 0/4
Map 1: 0(+2)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+2)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 6(+1)/7  Reducer 2: 0(+1)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 2(+1)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 4/4
OK
```

```
accessories.bag
appliances.environment.vacuum
appliances.personal.hair_cutter
sport.diving

apparel.glove
furniture.bathroom.bath
furniture.living_room.cabinet
stationery.cartrige
accessories.cosmetic_bag
appliances.environment.air_conditioner
furniture.living_room.chair
Time taken: 29.888 seconds, Fetched: 12 row(s)
```

Note: change empty values to NULL values:

```
hive> Alter table store_data set tblproperties('serialization.null.format' = "");
OK
Time taken: 0.058 seconds

hive> select distinct brand from store_data;
Query ID = hadoop_20221001184039_a2fb9b3b-1aa4-42c8-a112-277276251c2f
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0012)

Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+2)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+3)/7  Reducer 2: 0/4
Map 1: 3(+2)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+2)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 6(+1)/7  Reducer 2: 0(+1)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 1(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 4/4
OK
```

```
almea
artex
barbie
batiste
beautix
beautyblender
biore
blise
blixz
browxenna
busch
concept
cutrin
deoproce
dessata
domix
embryolisse
emil
enigma
entity
eos
f.o.x
fancy
farmavita
fedua
freshbubble
gena
glysolid
greymy
happyfons
haruyama
helloganic
i-laq
ibd
ikoo
jaguar
kaaral
kares
keen
laboratorium
lakme
lianail
lunaris
macadamia
mane
markell
masura
max
miskin
missha
moyou
nagaraku
nefertiti
nitrile
nova
orly
philips
provoc
pueen
shik
siberina
skinlite
skipofit
smart
soleo
strong
thuya
uno
uskusi
yoko
zab
zinger

airnails
andrea
balbcare
beauugreen
benovy
bergamo
bosnic
cnd
cristalinas
cuccio
de.lux
dermacol
dewal
enjoy
essie
estelare
farmona
farmstay
freedecor
godefroy
grace
grattol
ingarden
inoface
invisibobble
irisk
italwax
jas
kapous
kims
kiss
kocostar
koreatida
labay
ladykin
lsanic
marutaka-foot
matreshka
metzger
neoleor
oniq
opi
profepil
radius
rasyan
refectocil
rosi
roubloff
severina
shary
skinity
solomeya
staleks
sunuv
supertan
tannymaxx
tazol
tertio
vilenta
vl-gel
weaver
ypsed
yu-r
ardell
art-visage
australis
bioaqua
carmex
consly
coocla
dr.gloderm
ecocraft
ecolab
egomania
ellips
elskin
enas
esquire
fly
frozen
gehwol
inm
insight
joico
juno
kamill
kaypro
keune
konad
lamixx
levissime
likato
limoni
lovely
marathon
mavala
meisterwerk
mielle
milv
naomi
nirvel
nitrimax
osmo
ovale
plazan
pole
profhenna
protokeratin
rocknailstar
runail
sophin
tosowoong
trind
uralsoap
voesh
vosev
aura
beauty-free
bespecial
binacil
biofollica
bluesky
bodipure
bodyton
bpw.style
candy
chi
coifin
cosima
cosmoprofi
coxir
cruset
depilflax
dermal
dizao
dorena
elizavecca
estel
eunyul
finish
foamie
footlogix
igrobeauty
jessnail
kerasys
kinetics
koelcia
koelf
kosmekka
lador
laiseven
latinoil
lebelage
levrana
litaline
lowence
matrix
naturmed
parachute
petitfee
pnb
polarus
riche
s.care
sanoto
sawa
shifei
sun
swarovski
treaclemoon
veraclara
zeitun
Time taken: 28.999 seconds,
Fetched: 245 row(s)
```

QUERY 5: Find the total number of products available under each category:

```
hive> select category_code as category, count(product_id) as products from store_data where category_code is
not null group by category_code;
Query ID = hadoop_20221001184225_5fd72530-340c-48d8-a143-3b6f5dc4af3f
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0012)

Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0/7      Reducer 2: 0/4
Map 1: 0(+1)/7  Reducer 2: 0/4
Map 1: 0(+2)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 0(+3)/7  Reducer 2: 0/4
Map 1: 1(+2)/7  Reducer 2: 0/4
Map 1: 1(+3)/7  Reducer 2: 0/4
Map 1: 2(+3)/7  Reducer 2: 0/4
Map 1: 3(+3)/7  Reducer 2: 0/4
Map 1: 4(+3)/7  Reducer 2: 0/4
Map 1: 5(+2)/7  Reducer 2: 0/4
Map 1: 5(+2)/7  Reducer 2: 0(+1)/4
Map 1: 6(+1)/7  Reducer 2: 0(+2)/4
Map 1: 7/7      Reducer 2: 0(+3)/4
Map 1: 7/7      Reducer 2: 1(+2)/4
Map 1: 7/7      Reducer 2: 2(+2)/4
Map 1: 7/7      Reducer 2: 3(+1)/4
Map 1: 7/7      Reducer 2: 4/4
OK
```

```
accessories.bag 11681
appliances.environment.vacuum   59761
appliances.personal.hair_cutter 1643
sport.diving    2
apparel.glove   18232
furniture.bathroom.bath 9857
furniture.living_room.cabinet   13439
stationery.cartrige     26722
accessories.cosmetic_bag        1248
appliances.environment.air_conditioner  332
furniture.living_room.chair     308
Time taken: 29.315 seconds, Fetched: 11 row(s)
```

QUERY 6: which brand had the maximum sales in october and november combined?

```
hive> select brand, round(sum(price),2) as max_sales from store_data where brand is not null and event_type ='purchase'
group by brand order by max_sales desc limit 1;
Query ID = hadoop_20221001184631_900033c2-9c81-4d76-a8a4-1665f50c6fea
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0012)

Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+1)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 1(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 1(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 2(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 2(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 3(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 4(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 4(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 5(+2)/7  Reducer 2: 0(+1)/2      Reducer 3: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+1)/2      Reducer 3: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+2)/2      Reducer 3: 0/1
Map 1: 7/7      Reducer 2: 0(+2)/2      Reducer 3: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 0(+1)/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 1/1
OK
runail  148297.94
Time taken: 31.923 seconds, Fetched: 1 row(s)
```

QUERY 7.1: which brands increased their sales from October to November?

```
hive> With high_brand as ( select brand, month(event_time) as mnth,sum(price) as sales,dense_rank()
over(partition by brand order by sum(price) desc) as rank from store_data where brand is not null and
event_type = 'purchase' group by brand, month(event_time)order by brand, mnth) select brand from high_brand
where rank = 1 and mnth=11;
Query ID = hadoop_20221001185305_954d55e1-87da-4f8b-9c59-af52d67af1f5
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_1664627482233_0013)

Map 1: -/-      Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+1)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 1(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 1(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 2(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 5(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 5(+2)/7  Reducer 2: 0(+1)/2      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+1)/2      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+2)/2      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 7/7      Reducer 2: 0(+2)/2      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 0(+1)/1      Reducer 4: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 1/1  Reducer 4: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 1/1  Reducer 4: 0(+1)/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 1/1  Reducer 4: 1/1
OK
```

```
airnails
art-visage
artex
aura
balbcare
barbie
batiste
beautix
beauty-free
beautyblender
beauugreen
benovy
binacil
bioaqua
biore
blixz
bluesky
bodyton
bpw.style
browxenna
candy
carmex
chi
coifin
concept
cosima
cosmoprofi
cristalinas
cutrin
de.lux
deoproce
depilflax
dewal
dizao
domix
ecocraft
ecolab
egomania
elizavecca
ellips
elskin
enjoy
entity
eos
estel
estelare
f.o.x
farmavita
farmona
fedua
finish
fly
foamie
freedecor
freshbubble
gehwol
glysolid
godefroy
grace
grattol
greymy
happyfons
haruyama
helloganic
igrobeauty
ingarden
inm
insight
irisk
italwax
jaguar
jas
jessnail
joico
juno
kaaral
kamill
kapous
kares
kaypro
keen
kerasys
kims
kinetics
kiss
kocostar
koelcia
koelf
konad
kosmekka
laboratorium
lador
ladykin
latinoil
levissime
levrana
lianail
likato
limoni
lovely
lowence
mane
marathon
markell
marutaka-foot
masura
matreshka
matrix
mavala
metzger
milv
miskin
missha
moyou
nagaraku
naomi
nefertiti
neoleor
nirvel
nitrile
oniq
orly
Osmo
ovale
plazan
polarus
profepil
profhenna
protokeratin
provoc
rasyan
refectocil
rosi
roubloff
runail
s.care
sanoto
severina
shary
shik
skinity
skinlite
smart
soleo
solomeya
sophin
staleks
strong
supertan
swarovski
tertio
treaclemoon
trind
uno
uskusi
veraclara
vilenta
yoko
yu-r
zeitun
Time taken: 42.015 seconds, Fetched:
160 row(s)
```

QUERY 7.2: which brands increased their sales from October to November?

```
hive> With high_brand as (select brand, month(event_time) as mnth,sum(price) as sales,dense_rank()
over(partition by brand order by sum(price) desc) as rank from oct_data_1 where brand is not null and
event_type = 'purchase' group by brand, month(event_time) order by brand, mnth) select brand from
high_brand where rank = 1 and mnth=11;
Query ID = hadoop_20221001185550_5cefe304-ae2f-40be-a126-a35329223d6c
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0013)

Map 1: 0/3      Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0/3      Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0/3      Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+1)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+2)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 0(+3)/3  Reducer 2: 0/1  Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 2(+1)/3  Reducer 2: 0(+1)/1      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 3/3      Reducer 2: 0(+1)/1      Reducer 3: 0/1  Reducer 4: 0/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 0(+1)/1      Reducer 4: 0/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 1/1  Reducer 4: 0/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 1/1  Reducer 4: 0(+1)/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 1/1  Reducer 4: 1/1
OK
```

```
airnails
art-visage
artex
aura
balbcare
barbie
batiste
beautix
beauty-free
beautyblender
beauugreen
benovy
binacil
bioaqua
biore
blixz
bluesky
bodyton
bpw.style
browxenna
candy
carmex
chi
coifin
concept
cosima
cosmoprofi
cristalinas
cutrin
de.lux
deoproce
depilflax
dewal
dizao
domix
ecocraft
ecolab
egomania
elizavecca
ellips
elskin
enjoy
entity
eos
estel
estelare
f.o.x
farmavita
farmona
fedua
finish
fly
foamie
freedecor
freshbubble
gehwol
glysolid
godefroy
grace
grattol
greymy
happyfons
haruyama
helloganic
igrobeauty
ingarden
inm
insight
irisk
italwax
jaguar
jas
jessnail
joico
juno
kaaral
kamill
kapous
kares
kaypro
keen
kerasys
kims
kinetics
kiss
kocostar
koelcia
koelf
konad
kosmekka
laboratorium
lador
ladykin
latinoil
Levissime
levrana
lianail
likato
limoni
lovely
lowence
mane
marathon
markell
marutaka-foot
masura
matreshka
matrix
mavala
Metzger
milv
miskin
missha
moyou
nagaraku
naomi
nefertiti
neoleor
nirvel
nitrile
oniq
orly
osmo
ovale
plazan
polarus
profepil
profhenna
protokeratin
provoc
rasyan
refectocil
rosi
roubloff
runail
s.care
sanoto
severina
shary
shik
skinity
skinlite
smart
soleo
solomeya
sophin
staleks
strong
supertan
swarovski
tertio
treaclemoon
trind
uno
uskusi
veraclara
vilenta
yoko
yu-r
zeitun
Time taken: 25.854 seconds, Fetched:
161 row(s)
```

QUERY 8: your company wants to rewards the top 10 users of its website with a Golden customre plan. write a query to generate a list of top 10 users who spend the most.

```
hive> select user_id, sum(price) as purchase, dense_rank() over ( order by sum(price) desc) as rank from store_data where
event_type='purchase' group by user_id limit 10;
Query ID = hadoop_20221001185656_4ebcddf9-dc4c-4e07-a49f-649cb43830f4
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0013)

Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0/7      Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 0(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 1(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 2(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 3(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 4(+2)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 4(+3)/7  Reducer 2: 0/2  Reducer 3: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+1)/2      Reducer 3: 0/1
Map 1: 6(+1)/7  Reducer 2: 0(+2)/2      Reducer 3: 0/1
Map 1: 7/7      Reducer 2: 0(+2)/2      Reducer 3: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 0/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 0(+1)/1
Map 1: 7/7      Reducer 2: 2/2  Reducer 3: 1/1
OK
557790271       2715.8699957430363      1
150318419       1645.970008611679       2
562167663       1352.8499938696623      3
531900924       1329.4499949514866      4
557850743       1295.4800310581923      5
522130011       1185.3899966478348      6
561592095       1109.700007289648       7
431950134       1097.5900000333786      8
566576008       1056.3600097894669      9
521347209       1040.9099964797497      10
Time taken: 32.976 seconds, Fetched: 10 row(s)
```

```
hive> select user_id, sum(price) as purchase, dense_rank() over(order by sum(price)desc) as rank from
oct_data_1 group by user_id limit 10;
Query ID = hadoop_20221001185735_2cae4349-8967-426f-b8cc-6d4512eea3ed
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1664627482233_0013)

Map 1: 0(+1)/3  Reducer 2: 0/1  Reducer 3: 0/1
Map 1: 0(+2)/3  Reducer 2: 0/1  Reducer 3: 0/1
Map 1: 1(+1)/3  Reducer 2: 0/1  Reducer 3: 0/1
Map 1: 1(+1)/3  Reducer 2: 0(+1)/1      Reducer 3: 0/1
Map 1: 2(+0)/3  Reducer 2: 0(+1)/1      Reducer 3: 0/1
Map 1: 2(+1)/3  Reducer 2: 0(+1)/1      Reducer 3: 0/1
Map 1: 2(+1)/3  Reducer 2: 0(+1)/1      Reducer 3: 0/1
Map 1: 3/3      Reducer 2: 0(+1)/1      Reducer 3: 0/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 0/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 0(+1)/1
Map 1: 3/3      Reducer 2: 1/1  Reducer 3: 1/1
OK
557790271       2715.8699957430363      1
150318419       1645.970008611679       2
562167663       1352.8499938696623      3
531900924       1329.4499949514866      4
557850743       1295.4800310581923      5
522130011       1185.3899966478348      6
561592095       1109.700007289648       7
431950134       1097.5900000333786      8
566576008       1056.3600097894669      9
521347209       1040.9099964797497      10
Time taken: 10.247 seconds, Fetched: 10 row(s)
hive>
```

Final conclusion:

1.The performance wise partition is effective for low volume data,
Above we observed that performance rate increase when we use partitions.

2.For larger data creating a bucketing give us 20% to 30% better querry performance
then a non bucket table.

3. Based on the above data views , cart, event_type are more comparable to purchases.

4.The Total revenue is higher in November then October.

5. Highest number of products available under appliances, environment, vaccume category.

6. Runail brand has highest sales compared with other brands.

7. In general 43% brands are successfull in increasing their sales from October to November.

8.The user_id 557790271 spent higher in two months.