Nikhil Golambade
Yashaswini M

Hive Case Study

Problem Statement: The tech companies are exploring different ways to improve their sales by analysing customer behaviour and gaining insights about product trends, due to the increasing popularity in online sales. Therefore, as a big data analyst, we are expected to extract data and gather insights from a real-life data set of an e-commerce company.

Objective: Need to analyse and gain insights about the clickstream data from a website so that we can extract insights about the customers behaviour. The steps involved in the entire process are as follows:

The implementation phase can be divided into the following parts:

- Copying the data set into the HDFS:
    - Launch an EMR cluster that utilizes the Hive services, and
    - Move the data from the S3 bucket into the HDFS

- Creating the database and launching Hive queries on your EMR cluster:
    - Create the structure of your database,
    - Use optimized techniques to run your queries as efficiently as possible
    - Show the improvement of the performance after using optimization on any single query.
    - Run Hive queries to answer the questions given below.

- Cleaning up
    - Drop your database, and
    - Terminate your cluster

Nikhil Golambade
Yashaswini M

LAUNCH THE EMR CLUSTER.

In order to launch an EMR Cluster, the following steps were followed:

KEY-PAIR CREATION:

Creating Key Pair with .pem /.ppk file format directly to use with putty.

Click on Create key pair

Nikhil Golambade
Yashaswini M

HiveCaseStudy-KeyPair successfully created and downloaded

(Refer above screenshot).

Nikhil Golambade
Yashaswini M

S3 BUCKET:

To Store the data – Click on “Create Bucket”

Creating “hivecasestudy-nikhil-yashaswini” with all default options

Nikhil Golambade
Yashaswini M

Bucket Successfully created.

Click On Upload

Add files 2019-Oct.csv and 2019-Nov.csv

Click on Upload.

Nikhil Golambade
Yashaswini M

Successfully uploaded the 2019 October and 2019 November csv files to S3 bucket

Nikhil Golambade
Yashaswini M

EMR CLUSTER CREATION:

Click on “Create cluster” button to create the EMR cluster

Creating cluster with advanced options

Software and Steps page: Changed the Release from **emr-5.33.0** to “**emr-5.29.0**”

Nikhil Golambade
Yashaswini M

Hardware page: Changing the Master and Core nodes from **m5.xlarge** to "**m4.large**"

General Cluster Settings page: Giving the name to cluster "**HiveCaseStudy**"

Security page: Changing the EC2 key pair option to our created key pair – "**HiveCaseStudy-KeyPair**"

Click on "Create Cluster" button

Nikhil Golambade
Yashaswini M

Cluster is ready with status “Waiting”

Both the Master and Core nodes are running.

Nikhil Golambade
Yashaswini M

Click on **Security Groups** Of Master Node and Change the inbound rule

Click On Edit inbound rules

Nikhil Golambade
Yashaswini M

Add rule

Select SSH And select IP address to Anywhere

Nikhil Golambade
Yashaswini M

Successfully save the SSH rule to the inbound rules

Nikhil Golambade
Yashaswini M

CONNECT TO MASTER NODE

Open the putty and enter the Host Name highlighted in Screen Shot and navigate to Connection > SSH > Auth then browse and select the private key, which we creating initially.

Click Accept

Nikhil Golambade
Yashaswini M

EMR CLI is launched

Verifying the services that are running on Hadoop cluster with command

sudo initctl list

Nikhil Golambade
Yashaswini M

hadoop fs -ls /

All the above are inbuilt directories in HDFS.

CREATING A NEW DIRECTORY FOR HIVE CASE STUDY:

hadoop fs -mkdir /Hive_assignment
hadoop fs -ls /

New directory is successfully created

Nikhil Golambade
Yashaswini M

LOADING THE DATA FROM S3 BUCKET to HDFS:

hadoop distcp 's3://hivecasestudy-nikhil-yashaswini/2019-Oct.csv' /Hive_assignment/2019-Oct.csv

```
hadoop@ip-172-31-72-89:~
[hadoop@ip-172-31-72-89 ~]$ hadoop distcp 's3://hivecasestudy-nikhil-yashaswini/
2019-Oct.csv' /Hive_assignment/2019-Oct.csv
21/08/30 09:18:48 INFO tools.DistCp: Input Options: DistCpOptions{atomicCommit=f
alse, syncFolder=false, deleteMissing=false, ignoreFailures=false, overwrite=fal
se, skipCRC=false, blocking=true, numListstatusThreads=0, maxMaps=20, mapBandwid
th=100, sslConfigurationFile='null', copyStrategy='uniformsize', preserveStatus=
[], preserveRawXattrs=false, atomicWorkPath=null, logPath=null, sourceFileListin
g=null, sourcePaths=[s3://hivecasestudy-nikhil-yashaswini/2019-Oct.csv], targetP
ath=/Hive_assignment/2019-Oct.csv, targetPathExists=false, filtersFile='null'}
```

```
hadoop@ip-172-31-72-89:~
        File Input Format Counters
                Bytes Read=234
        File Output Format Counters
                Bytes Written=0
        DistCp Counters
                Bytes Copied=482542278
                Bytes Expected=482542278
                Files Copied=1
[hadoop@ip-172-31-72-89 ~]$
```

hadoop distcp 's3://hivecasestudy-nikhil-yashaswini/2019-Nov.csv' /Hive_assignment/2019-Nov.csv

```
hadoop@ip-172-31-72-89:~
[hadoop@ip-172-31-72-89 ~]$ hadoop distcp 's3://hivecasestudy-nikhil-yashaswini/
2019-Nov.csv' /Hive_assignment/2019-Nov.csv
21/08/30 09:23:47 INFO tools.DistCp: Input Options: DistCpOptions{atomicCommit=f
alse, syncFolder=false, deleteMissing=false, ignoreFailures=false, overwrite=fal
se, skipCRC=false, blocking=true, numListstatusThreads=0, maxMaps=20, mapBandwid
th=100, sslConfigurationFile='null', copyStrategy='uniformsize', preserveStatus=
[], preserveRawXattrs=false, atomicWorkPath=null, logPath=null, sourceFileListin
g=null, sourcePaths=[s3://hivecasestudy-nikhil-yashaswini/2019-Nov.csv], targetP
ath=/Hive_assignment/2019-Nov.csv, targetPathExists=false, filtersFile='null'}
```

```
hadoop@ip-172-31-72-89:~
        File Input Format Counters
                Bytes Read=234
        File Output Format Counters
                Bytes Written=0
        DistCp Counters
                Bytes Copied=545839412
                Bytes Expected=545839412
                Files Copied=1
[hadoop@ip-172-31-72-89 ~]$
```

Nikhil Golambade
Yashaswini M

Inspecting the table data to know which columns are available before creating the hive table with command.

hadoop fs -cat /Hive_assignment/2019-Oct.csv | head

```
hadoop@ip-172-31-72-89:~
[hadoop@ip-172-31-72-89 ~]$ hadoop fs -cat /Hive_assignment/2019-Oct.csv | head
event_time,event_type,product_id,category_id,category_code,brand,price,user_id,u
ser_session
2019-10-01 00:00:00 UTC,cart,5773203,1487580005134238553,,runail,2.62,463240011,
26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:03 UTC,cart,5773353,1487580005134238553,,runail,2.62,463240011,
26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:07 UTC,cart,5881589,2151191071051219817,,lovely,13.48,429681830
,49e8d843-adf3-428b-a2c3-fe8bc6a307c9
2019-10-01 00:00:07 UTC,cart,5723490,1487580005134238553,,runail,2.62,463240011,
26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:15 UTC,cart,5881449,1487580013522845895,,lovely,0.56,429681830,
49e8d843-adf3-428b-a2c3-fe8bc6a307c9
2019-10-01 00:00:16 UTC,cart,5857269,1487580005134238553,,runail,2.62,430174032,
73dea1e7-664e-43f4-8b30-d32b9d5af04f
2019-10-01 00:00:19 UTC,cart,5739055,1487580008246412266,,kapous,4.75,377667011,
81326ac6-daa4-4f0a-b488-fd0956a78733
2019-10-01 00:00:24 UTC,cart,5825598,1487580009445982239,,,0.56,467916806,2f5b55
46-b8cb-9ee7-7ecd-84276f8ef486
2019-10-01 00:00:25 UTC,cart,5698989,1487580006317032337,,,1.27,385985999,d30965
e8-1101-44ab-b45d-cc1bb9fae694
```

hadoop fs -cat /Hive_assignment/2019-Nov.csv | head

```
hadoop@ip-172-31-72-89:~
[hadoop@ip-172-31-72-89 ~]$ hadoop fs -cat /Hive_assignment/2019-Nov.csv | head
event_time,event_type,product_id,category_id,category_code,brand,price,user_id,u
ser_session
2019-11-01 00:00:02 UTC,view,5802432,1487580009286598681,,,0.32,562076640,09fafd
6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:09 UTC,cart,5844397,1487580006317032337,,,2.38,553329724,206721
6c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:10 UTC,view,5837166,1783999064103190764,,pnb,22.22,556138645,57
ed222e-a54a-4907-9944-5a875c2d7f4f
2019-11-01 00:00:11 UTC,cart,5876812,1487580010100293687,,jessnail,3.16,56450666
6,186c1951-8052-4b37-adce-dd9644b1d5f7
2019-11-01 00:00:24 UTC,remove_from_cart,5826182,1487580007483048900,,,3.33,5533
29724,2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:24 UTC,remove_from_cart,5826182,1487580007483048900,,,3.33,5533
29724,2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:25 UTC,view,5856189,1487580009026551821,,runail,15.71,562076640
,09fafd6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:32 UTC,view,5837835,1933472286753424063,,,3.49,514649199,432a4e
95-375c-4b40-bd36-0fc039e77580
2019-11-01 00:00:34 UTC,remove_from_cart,5870838,1487580007675986893,,milv,0.79,
429913900,2f0bff3c-252f-4fe6-afcd-5d8a6a92839a
```

Nikhil Golambade
Yashaswini M

MOVING TO HIVE:

hive

```
hadoop@ip-172-31-72-89:~
[hadoop@ip-172-31-72-89 ~]$ hive

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.
properties Async: false
hive>
```

show databases;

```
hadoop@ip-172-31-72-89:~
hive> show databases;
OK
default
Time taken: 0.692 seconds, Fetched: 1 row(s)
hive>
```

CREATING NEW DATABASE “casestudy”:

create database if not exists casestudy;

```
hadoop@ip-172-31-77-138:~
hive> create database if not exists casestudy;
OK
Time taken: 0.069 seconds
```

Show databases;
describe database casestudy;

```
hadoop@ip-172-31-77-138:~
hive> show databases;
OK
casestudy
default
Time taken: 0.012 seconds, Fetched: 2 row(s)
hive> describe database casestudy;
OK
casestudy               hdfs://ip-172-31-77-138.ec2.internal:8020/user/hive/ware
house/casestudy.db      hadoop  USER
Time taken: 0.021 seconds, Fetched: 1 row(s)
hive>
```

CREATING AN EXTERNAL TABLE IN HIVE:

CREATE EXTERNAL TABLE IF NOT EXISTS retail (event_time timestamp, event_type string,
product_id string, category_id string, category_code string, brand string, price decimal(10,3), user_id bigint,user_session string) ROW FORMAT SERDE
'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH
SERDEPROPERTIES ("separatorChar" = "," , "quoteChar" = "\"", "escapeChar" = "\\") stored as textfile
LOCATION '/Hive_assignment' TBLPROPERTIES ("skip.header.line.count"="1") ;

Nikhil Golambade
Yashaswini M

```
hive>
    > CREATE EXTERNAL TABLE IF NOT EXISTS retail (event_time timestamp, event_type string,
    > product_id string, category_id string, category_code string, brand string, price decimal(10,3), user_id bigint,
    > user_session string) ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH
    > SERDEPROPERTIES ("separatorChar" = "," , "quoteChar" = "\"", "escapeChar" = "\\") stored as textfile
    > LOCATION '/Hive_assignment' TBLPROPERTIES ("skip.header.line.count"="1") ;
OK
Time taken: 0.861 seconds
hive>
```

describe retail;

```
hive> describe retail;
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: 0.188 seconds, Fetched: 9 row(s)
hive>
```

load data inpath '/Hive_assignment/2019-Oct.csv' into table retail;

```
hive>
    > load data inpath '/Hive_assignment/2019-Oct.csv' into table retail;
Loading data to table default.retail
OK
Time taken: 2.068 seconds
hive>
```

load data inpath '/Hive_assignment/2019-Nov.csv' into table retail;

```
hive> load data inpath '/Hive_assignment/2019-Nov.csv' into table retail;
Loading data to table default.retail
OK
Time taken: 1.707 seconds
hive>
```

Nikhil Golambade
Yashaswini M

VERIFYING THE PERFORMANCE :

select * from retail where month(event_time)=11 limit 5;

```
hive> select * from retail where month(event_time)=11 limit 5;
OK
2019-11-01 00:00:02 UTC view    5802432 1487580009286598681                    0
.32     562076640       09fafd6c-6c99-46b1-834f-33527f4de241
2019-11-01 00:00:09 UTC cart    5844397 1487580006317032337                    2
.38     553329724       2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:10 UTC view    5837166 1783999064103190764             pnb    2
2.22    556138645       57ed222e-a54a-4907-9944-5a875c2d7f4f
2019-11-01 00:00:11 UTC cart    5876812 1487580010100293687             jessnail
3.16    564506666       186c1951-8052-4b37-adce-dd9644b1d5f7
2019-11-01 00:00:24 UTC remove_from_cart        5826182 1487580007483048900    3
.33     553329724       2067216c-31b5-455d-a1cc-af0575a34ffb
Time taken: 2.43 seconds, Fetched: 5 row(s)
```

select * from retail where month(event_time)=10 limit 5;

```
hive> select * from retail where month(event_time)=10 limit 5;
OK
2019-10-01 00:00:00 UTC cart    5773203 1487580005134238553          runail  2.62    463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:03 UTC cart    5773353 1487580005134238553          runail  2.62    463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:07 UTC cart    5881589 2151191071051219817          lovely  13.48   429681830       49e8d843-adf3-428b-a2c3-fe8bc6a307c9
2019-10-01 00:00:07 UTC cart    5723490 1487580005134238553          runail  2.62    463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885
2019-10-01 00:00:15 UTC cart    5881449 1487580013522845895          lovely  0.56    429681830       49e8d843-adf3-428b-a2c3-fe8bc6a307c9
Time taken: 0.217 seconds, Fetched: 5 row(s)
hive>
```

Checking retail table before Optimizing:

SELECT SUM(price) FROM retail WHERE MONTH(event_time)=10 AND event_type='purchase' ;

```
hive> SELECT SUM(price) FROM retail WHERE MONTH(event_time)=10 AND event_type='purchase' ;
Query ID = hadoop_20210831070346_e5875aba-6cbe-4cd4-b211-c79eff9f09d9
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0009)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........  container     SUCCEEDED      2          2        0        0       0       0
Reduce 2 ......  container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 60.50 s
--------------------------------------------------------------------------------
OK
1211538.4299997438
Time taken: 61.217 seconds, Fetched: 1 row(s)
hive>
```

Time Taken to execute the above query is 61.217 sec

This is very high. Hence, to reduce this execution time, we will dynamically partition the table “retail” and add buckets to create an optimised table.

Nikhil Golambade
Yashaswini M

APPLYING OPTIMIZATION TECHNIQUES - PARTITIONING AND BUCKETING:

Below commands are to enable the dynamic partitioning and bucketing
hive> set hive.exec.dynamic.partition.mode = nonstrict;
hive> set hive.exec.dynamic.partition = true;
hive> set hive.enforce.bucketing = true;

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

Partition on : event_type (there are 4 types and all questions are related to 'purchase')

CREATE EXTERNAL TABLE IF NOT EXISTS retail_1 (event_time timestamp, product_id string,

category_id string, category_code string, brand string, price decimal(10,3), user_id bigint, user_session

string) PARTITIONED BY(event_type string) CLUSTERED BY (user_id) INTO 5 buckets ROW FORMAT

SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS textfile ;

```
hive> CREATE EXTERNAL TABLE IF NOT EXISTS retail_1 (event_time timestamp, produc
t_id string,
    > category_id string, category_code string, brand string, price decimal(10,3
), user_id bigint, user_session
    > string) PARTITIONED BY(event_type string) CLUSTERED BY (user_id) INTO 5 bu
ckets ROW FORMAT
    > SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS textfile ;
OK
Time taken: 0.785 seconds
hive>
```

describe retail_1;

```
hive> describe retail_1;
OK
event_time              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
event_type              string

# Partition Information
# col_name              data_type               comment

event_type              string
Time taken: 0.495 seconds, Fetched: 14 row(s)
hive>
```

Nikhil Golambade
Yashaswini M

INSERTING THE DATA INTO NEWLY CREATED OPTIMIZED TABLE (retail_1) FROM EXISTING TABLE(retail):

INSERT INTO TABLE retail_1 PARTITION (event_type) SELECT event_time, product_id, category_id, category_code, brand, price, user_id, user_session, event_type FROM retail;

```
hive> INSERT INTO TABLE retail_1 PARTITION (event_type) SELECT event_time, product_id,
    > category_id, category_code, brand, price, user_id, user_session, event_type FROM retail;
Query ID = hadoop_20210831092206_8bdd7186-c789-433a-ad1b-89bfc0819e46
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0014)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      2          2        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      5          5        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 164.28 s
--------------------------------------------------------------------------------------------
Loading data to table default.retail_1 partition (event_type=null)

Loaded : 4/4 partitions.
         Time taken to load dynamic partitions: 0.478 seconds
         Time taken for adding to write entity : 0.003 seconds
OK
Time taken: 169.631 seconds
hive>
```

Output: Based on the above results, it partitioned into 4 Verifying the partitioned in the Hadoop file system

ANSWERING GIVEN QUESTIONS:

QUESTION 1:

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

Base table:

SELECT SUM(price) FROM retail WHERE MONTH(event_time)=10 AND event_type='purchase' ;

```
hive>
    > SELECT SUM(price) FROM retail WHERE MONTH(event_time)=10 AND
    > event_type='purchase' ;
Query ID = hadoop_20210831093619_66d75378-17aa-4e46-9cf0-80430fc2834c
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0014)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      2          2        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 59.94 s
--------------------------------------------------------------------------------------------
OK
1211538.4299997438
Time taken: 60.516 seconds, Fetched: 1 row(s)
hive>
```

Time taken is 60.516 sec.

Nikhil Golambade
Yashaswini M

Optimized table:

SELECT SUM(price) FROM retail_1 WHERE MONTH(event_time)=10 AND event_type='purchase' ;

```
hive> SELECT SUM(price) FROM retail_1 WHERE MONTH(event_time)=10 AND
    > event_type='purchase' ;
Query ID = hadoop_20210831092828_df1a59b9-b080-498e-ba9a-c40d83ac895a
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0014)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 22.88 s
--------------------------------------------------------------------------------
OK
1211538.4299998982
Time taken: 23.592 seconds, Fetched: 1 row(s)
hive>
```

Time taken with optimized table is 23.592 sec.

Insights:

- The total revenue generated based on Purchase made in the month of October is 1,211,538.43 /-
- Non-optimized table query took the execution time of 60.516 seconds whereas optimized table query took execution time of 23.592 seconds. We can see there is a significant drop in the execution time of the same query.
- Hence, optimized table gives better performance in execution time.

QUESTION 2:

Write a query to yield the total sum of purchases per month in a single output.
Base Query:
SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROM
retail WHERE event_type='purchase' GROUP BY MONTH(event_time) ;

```
hive> SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROM
    > retail WHERE event_type='purchase' GROUP BY MONTH(event_time) ;
Query ID = hadoop_20210831093356_fc44cf0e-fa17-4599-9d7c-59ea831ed3a8
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0014)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      2          2        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      3          3        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 56.61 s
--------------------------------------------------------------------------------
OK
10      1211538.4299997438      245624
11      1531016.900000122       322417
Time taken: 57.213 seconds, Fetched: 2 row(s)
hive>
```

Time taken 57.213 sec.

Nikhil Golambade
Yashaswini M

Optimized table:

SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROM
retail_1 WHERE event_type='purchase' GROUP BY MONTH(event_time) ;

```
hive>
    > SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROM
    > retail_1 WHERE event_type='purchase' GROUP BY MONTH(event_time) ;
Query ID = hadoop_20210831093138_fc536126-ec53-49b8-8943-24ca4f9f7eb3
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0014)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reduce 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 22.71 s
--------------------------------------------------------------------------------
OK
10      1211538.4299998982      245624
11      1531016.8999999384      322417
Time taken: 23.419 seconds, Fetched: 2 row(s)
hive>
```

Time taken is 23.419 sec.

Insights:

- Sum of purchases made in the month of October is 245624 and in the month of November 322417, which means number of purchases are increased in November month
- Non-optimized table query took the execution time of 57.213 seconds whereas optimized table query took execution time of 23.419 seconds. We can see there is a significant drop in the execution time of the same query.
- Hence, with proper partitioning and bucketing on table we can reduce execution time.

Using Optimized table from below questions onwards:

QUESTION 3:

Write a query to find the change in revenue generated due to purchases from October to November.

WITH diff AS (SELECT SUM(CASE WHEN date_format(event_time,'MM')=10 THEN price ELSE 0
END) AS October, SUM(CASE WHEN date_format(event_time,'MM')=11 THEN price ELSE 0 END) AS
November FROM retail_1 WHERE date_format(event_time,'MM') IN (10,11) AND event_type='purchase') SELECT October, November, (November - October) as Difference FROM diff ;

Nikhil Golambade
Yashaswini M

```
hive> WITH diff AS ( SELECT SUM(CASE WHEN date_format(event_time,'MM')=10 THEN price ELSE 0
    > END) AS October, SUM(CASE WHEN date_format(event_time,'MM')=11 THEN price ELSE 0 END) AS
    > November FROM retail_1 WHERE date_format(event_time,'MM') IN (10,11) AND
    > event_type='purchase') SELECT October, November, (November - October) as Difference FROM diff ;
Query ID = hadoop_20210831094425_8607df34-2371-40c0-b1cf-4341fbba87ee
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0015)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reduce 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 33.46 s
--------------------------------------------------------------------------------------------
OK
1211538.429999898       1531016.8999999384      319478.4700000405
Time taken: 38.094 seconds, Fetched: 1 row(s)
hive>
    >
```

Insights:

- Time taken to execute the query is 38.094 seconds
- Revenue increased in November by 319478.47 from October

QUESTION 4:

Find distinct categories of products. Categories with null category code can be ignored.

SELECT DISTINCT split(category_code,'\\.')[0] AS category FROM retail_1 WHERE split(category_code,'\\.')[0]<>'' ;

```
hive> SELECT DISTINCT split(category_code,'\\.')[0] AS category FROM retail_1 WHERE
    > split(category_code,'\\.')[0]<>'' ;
Query ID = hadoop_20210831095221_bc00b99d-e8bd-4825-b67e-b4f19124b4d9
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0015)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      4          4        0        0       0       0
Reduce 2 ...... container     SUCCEEDED      5          5        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 66.64 s
--------------------------------------------------------------------------------------------
OK
furniture
appliances
accessories
apparel
sport
stationery
Time taken: 67.402 seconds, Fetched: 6 row(s)
hive>
```

Insights:

- Time taken to execute the query is 67.402 seconds
- Total we got 6 distinct categories are – furniture, appliances, accessories, apparel, sport, stationary.

Nikhil Golambade
Yashaswini M

QUESTION 5:

Find the total number of products available under each category.

SELECT split(category_code,'\\.')[0] AS category, COUNT(product_id) AS prd FROM retail_1
GROUP BY split(category_code,'\\.')[0] ORDER BY prd DESC ;

```
hive> SELECT split(category_code,'\\.')[0] AS category, COUNT(product_id) AS prd FROM retail_1
    > GROUP BY split(category_code,'\\.')[0] ORDER BY prd DESC ;
Query ID = hadoop_20210831100709_4bbc97fc-d027-4f92-899b-463886afe125
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0017)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      4          4        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      5          5        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 64.92 s
--------------------------------------------------------------------------------
OK
        8594895
appliances      61736
stationery      26722
furniture       23604
apparel 18232
accessories     12929
sport   2
Time taken: 65.765 seconds, Fetched: 7 row(s)
hive>
```

Insights:

- Time taken to execute the query is 65.765 seconds
- Appliances are having highest number of products available with 61736 compared to other categories.
- Stationery (26722) and Furniture (23604) categories are almost equally registered with available ranges from 23000 to 27000.
- Sports category is least available with 2 products.

QUESTION 6:

Which brand had the maximum sales in October and November combined?

SELECT brand, SUM(price) AS Sales FROM retail_1 WHERE brand <>'' AND event_type='purchase'
GROUP BY brand ORDER BY Sales DESC LIMIT 1 ;

Nikhil Golambade
Yashaswini M

```
hive> SELECT brand, SUM(price) AS Sales FROM retail_1 WHERE brand <>'' AND event_type='purchase'
    > GROUP BY brand ORDER BY Sales DESC LIMIT 1 ;
Query ID = hadoop_20210831102843_fbb825a9-f62d-4839-b39d-7c5243166c57
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0019)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 22.15 s
--------------------------------------------------------------------------------------------
OK
runail  148297.93999999898
Time taken: 23.232 seconds, Fetched: 1 row(s)
hive>
```

Insights:

- Time taken to execute the query is 23.232 seconds
- Runail is the brand that has the highest sales in total of both the months October and November
- It seems that Runail brand has high popularity among cosmetic lovers and bringing in more
- products related to Runail brand could help in increasing their profit.

QUESTION 7:

Which brands increased their sales from October to November?

```
WITH Monthly_rev AS (
SELECT brand,
SUM(CASE WHEN date_format(event_time, 'MM')=10 THEN price ELSE 0 END) AS
Oct_rev,
SUM(CASE WHEN date_format(event_time, 'MM')=11 THEN price ELSE 0 END) AS
Nov_rev
FROM retail_1
WHERE event_type='purchase' AND date_format(event_time, 'MM') IN ('10', '11')
GROUP BY brand )
SELECT brand, Oct_rev, Nov_rev, Nov_rev-Oct_rev AS Sales_diff
FROM Monthly_rev
WHERE (Nov_rev - Oct_rev)>0
ORDER BY Sales_diff;
```

Nikhil Golambade
Yashaswini M

hadoop@ip-172-31-77-138:~

```
hive> WITH Monthly_rev AS (
    > SELECT brand,
    > SUM(CASE WHEN date_format(event_time, 'MM')=10 THEN price ELSE 0 END) AS Oct_rev,
    > SUM(CASE WHEN date_format(event_time, 'MM')=11 THEN price ELSE 0 END) AS Nov_rev
    > FROM retail_1
    > WHERE event_type='purchase' AND date_format(event_time, 'MM') IN ('10', '11')
    > GROUP BY brand )
    > SELECT brand, Oct_rev, Nov_rev, Nov_rev-Oct_rev AS Sales_diff
    > FROM Monthly_rev
    > WHERE (Nov_rev - Oct_rev)>0
    > ORDER BY Sales_diff;
Query ID = hadoop_20210831125016_2bcc7c15-84ab-4485-8bc5-ae783b9dc028
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0022)

--------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 40.52 s
--------------------------------------------------------------------------------------------
OK
ovale   2.54    3.1     0.56
cosima  20.229999999999997      20.93   0.7000000000000028
grace   100.91999999999999      102.61000000000001      1.6900000000000261
helloganic      0.0     3.1     3.1
skinity 8.88    12.440000000000001      3.5600000000000005
bodyton 1376.3400000000006      1380.6400000000003      4.299999999999727
moyou   5.71    10.280000000000001      4.570000000000001
neoleor 43.41   51.7    8.290000000000006
soleo   204.1999999999998       212.5299999999999       8.330000000000098
jaguar  1102.1100000000004      1110.65 8.539999999999736
tertio  236.16  245.79999999999995      9.639999999999958
fly     17.14   27.17   10.030000000000001
rasyan  18.799999999999997      28.939999999999998      10.14
deoproce        316.84  329.17  12.330000000000041
barbie  0.0     12.39   12.39
supertan        50.36999999999999       66.50999999999999       16.14
treaclemoon     163.37  181.49000000000004      18.120000000000033
kamill  63.00999999999999       81.49000000000002       18.480000000000032
juno    0.0     21.08   21.08
veraclara       50.11   71.21000000000001       21.10000000000001
glysolid        69.72999999999999       91.59   21.860000000000014
godefroy        401.22  425.11999999999995      23.89999999999992
binacil 0.0     24.259999999999998      24.259999999999998
blixz   38.95   63.400000000000006      24.450000000000003
```

Nikhil Golambade
Yashaswini M

hadoop@ip-172-31-77-138:~

```
levissime       2227.5000000000045      3085.310000000011       857.8100000000063
art-visage      2092.710000000002       2997.800000000007       905.0900000000051
ecolab  262.85  1214.3000000000002      951.4500000000002
nagaraku        4369.740000000011       5327.680000000006       957.939999999995
sanoto  157.14  1209.6799999999998      1052.54
markell 1768.7499999999984      2834.429999999999       1065.6800000000005
metzger 5373.450000000001       6457.159999999996       1083.7099999999955
de.lux  1659.6999999999957      2775.509999999989       1115.8099999999931
swarovski       1887.9299999999944      3043.1599999999908      1155.2299999999964
beauty-free     554.1700000000002       1782.860000000002       1228.6900000000019
zeitun  708.6600000000001       2009.6300000000006      1300.9700000000005
joico   705.52  2015.1  1309.58
severina        4775.879999999989       6120.47999999998        1344.5999999999904
irisk   45591.95999999944       46946.039999999615      1354.0800000001764
oniq    8425.409999999998       9841.650000000003       1416.2400000000052
levrana 2243.5599999999995      3664.100000000004       1420.5400000000045
roubloff        3491.3599999999997      4913.770000000005       1422.4100000000053
smart   4457.259999999995       5902.139999999995       1444.88
shik    3341.2000000000007      4839.720000000001       1498.5200000000004
domix   10472.049999999996      12009.169999999967      1537.1199999999717
artex   2730.64 4327.249999999998       1596.6099999999983
beautix 10493.949999999984      12222.949999999992      1729.0000000000073
milv    3904.9399999999796      5642.009999999875       1737.070000000008
masura  31266.079999998827      33058.46999999955       1792.3900000007234
f.o.x   6624.229999999996       8577.279999999995       1953.0499999999993
kapous  11927.159999999969      14093.080000000014      2165.9200000000455
concept 11032.139999999992      13380.399999999941      2348.2599999999493
estel   21756.749999999916      24142.670000000035      2385.9200000001183
kaypro  881.34  3268.6999999999994      2387.359999999999
benovy  409.6199999999999       3259.970000000001       2850.3500000000013
italwax 21940.239999999896      24799.369999999915      2859.130000000019
yoko    8756.91 11707.879999999986      2950.9699999999866
haruyama        9390.689999999879       12352.910000000069      2962.2200000001903
marathon        7280.750000000003       10273.099999999999      2992.349999999996
lovely  8704.379999999994       11939.05999999998       3234.6799999999857
bpw.style       11572.150000000083      14837.44000000017       3265.2900000000864
staleks 8519.730000000014       11875.610000000015      3355.880000000001
freedecor       3421.779999999996       7671.799999999959       4250.019999999963
runail  71539.279999999 76758.65999999984       5219.380000000849
polarus 6013.719999999999       11371.930000000004      5358.210000000005
cosmoprofi      8322.809999999994       14536.990000000042      6214.180000000048
jessnail        26287.840000000127      33345.23000000014       7057.390000000014
strong  29196.630000000005      38671.27000000002       9474.640000000014
ingarden        23161.389999999883      33566.210000000225      10404.820000000342
lianail 5892.8399999999865      16394.23999999996       10501.399999999976
uno     35302.030000000006      51039.75000000007       15737.720000000067
grattol 35445.53999999993       71472.71000000341       36027.17000000348
        474679.0600000175       619509.2400000119       144830.17999999435
Time taken: 41.3 seconds, Fetched: 161 row(s)
hive>
```

Insights:

- Here are some 161 brands with increment in the selling from October to November.
- ‘Grattol’ brand has the highest total increment i.e., 36,027 /- and ‘Ovale’ seems to have the least increment of 0.56 /- from October to November.
- Among all these brands lists, ‘Runail’ which was the best brand in terms of selling in October and November combined is also in the top 10 brands with high increment for October (71539.28) to November (76758.6) i.e., increment of total 5219.38.
- This implies that ‘Runail’ is the best and popular brand among all other brands within people.

Nikhil Golambade
Yashaswini M

QUESTION 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.

```
SELECT user_id, SUM(price) as Total_Expense
FROM retail_1
WHERE event_type='purchase'
GROUP BY user_id
ORDER BY Total_Expense DESC
LIMIT 10;
```

```
hive> SELECT user_id, SUM(price) as Total_Expense
    > FROM retail_1
    > WHERE event_type='purchase'
    > GROUP BY user_id
    > ORDER BY Total_Expense DESC
    > LIMIT 10;
Query ID = hadoop_20210831125517_8e30b1e5-4241-462f-8590-7b5b9b619599
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1630390382764_0022)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      3          3        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 24.41 s
--------------------------------------------------------------------------------
OK
557790271       2715.869999999991
150318419       1645.9700000000005
562167663       1352.85
531900924       1329.45
557850743       1295.4800000000005
522130011       1185.3899999999999
561592095       1109.7000000000005
431950134       1097.5899999999997
566576008       1056.3600000000004
521347209       1040.9099999999999
Time taken: 25.122 seconds, Fetched: 10 row(s)
hive>
```

Insights:

- Here is the list of the top 10 users (User_ids along with the amount spent) or buyers who have spent the most and could be rewarded with a Golden Customer plan to attract more people in the coming future.
- With the Optimized table the execution time reduced with proper partitioning and bucketing.
- Time taken to execute this query on optimized table is 25.122 seconds.

Nikhil Golambade
Yashaswini M

CLEANING UP:

Once the analysis is completed, deleting the database.

```
hive> show databases;
OK
casestudy
default
Time taken: 0.606 seconds, Fetched: 2 row(s)
hive> drop database casestudy;
OK
Time taken: 0.064 seconds
hive> show databases;
OK
default
Time taken: 0.022 seconds, Fetched: 1 row(s)
hive>
```

TERMINATION PROCESS:

After completing our analysis, we should terminate the EMR cluster.

Cluster terminated!!