Ecommerce Cosmetics Retail Hive Case Study

Mohammed Zuhad Afnan

PROBLEM STATEMENT

With online sales gaining popularity, tech companies are exploring ways to improve their sales by analysing customer behaviour and gaining insights about product trends. Furthermore, the websites make it easier for customers to find the products they require without much scavenging.

One of the most popular use cases of Big Data is in ecommerce companies such as Amazon or Flipkart. So, before we get into the details of the dataset, let us understand how ecommerce companies make use of these concepts to give customers product recommendations. This is done by tracking your clicks on their website and searching for patterns within them. This kind of data is called a clickstream data.

The clickstream data contains all the logs as to how you navigated through the website. It also contains other details such as time spent on every page, etc.

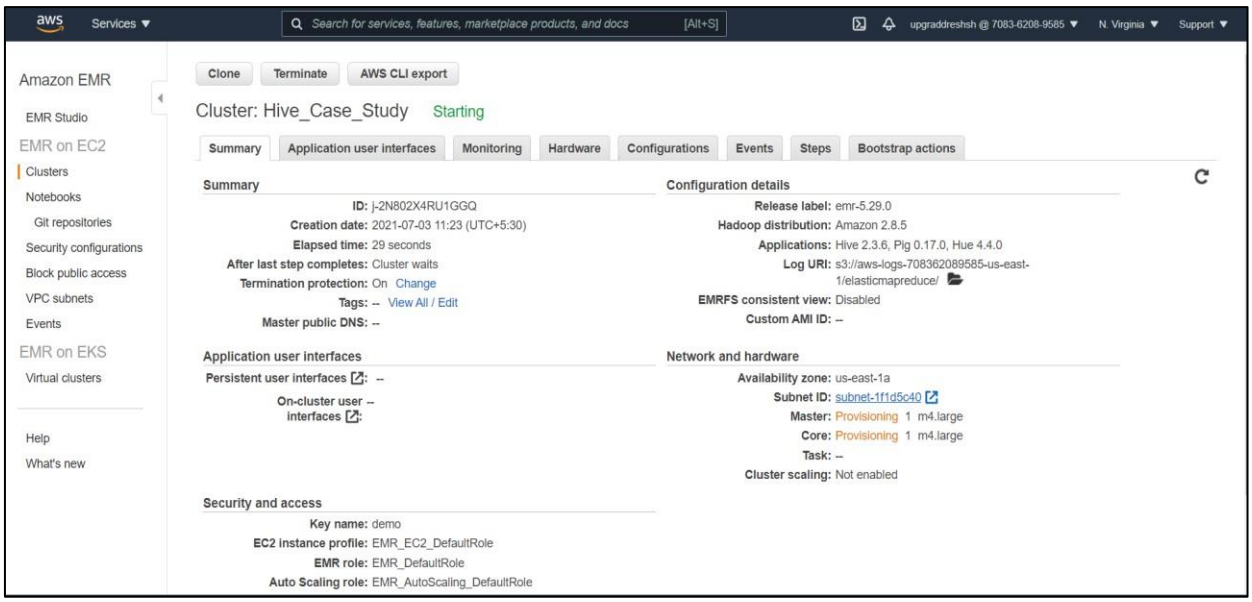
For this assignment, you will be working with a public clickstream dataset of a cosmetics store. Using this dataset, your job is to extract valuable insights.

DATA SOURCE

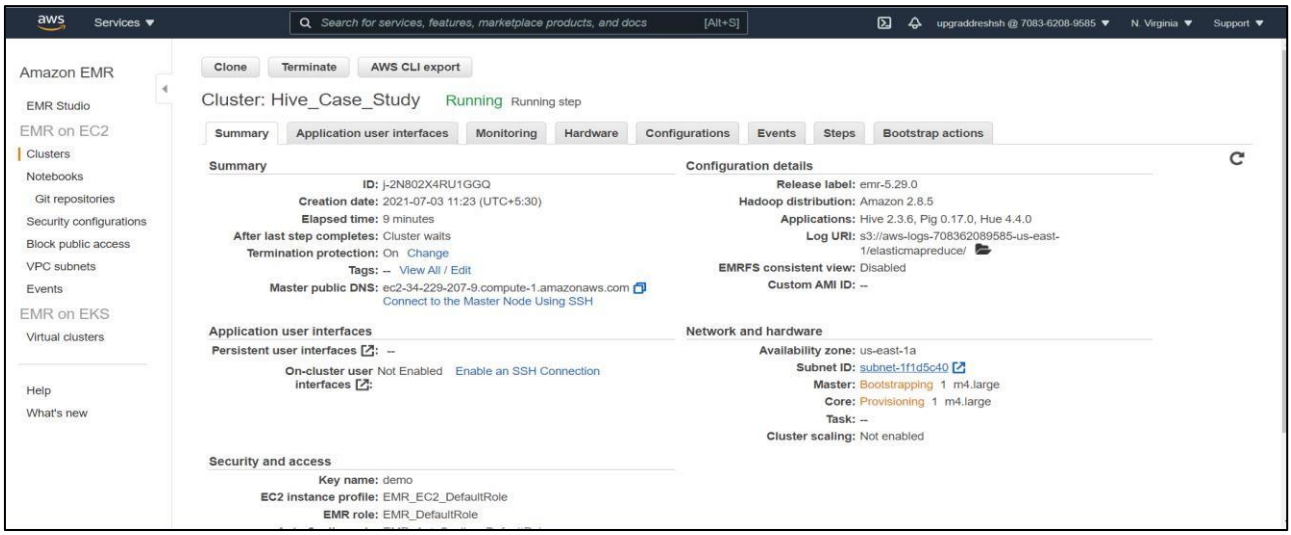
https://e-commerce-events-ml.s3.amazonaws.com/2019-Oct.csv
https://e-commerce-events-ml.s3.amazonaws.com/2019-Nov.csv

SOLUTION

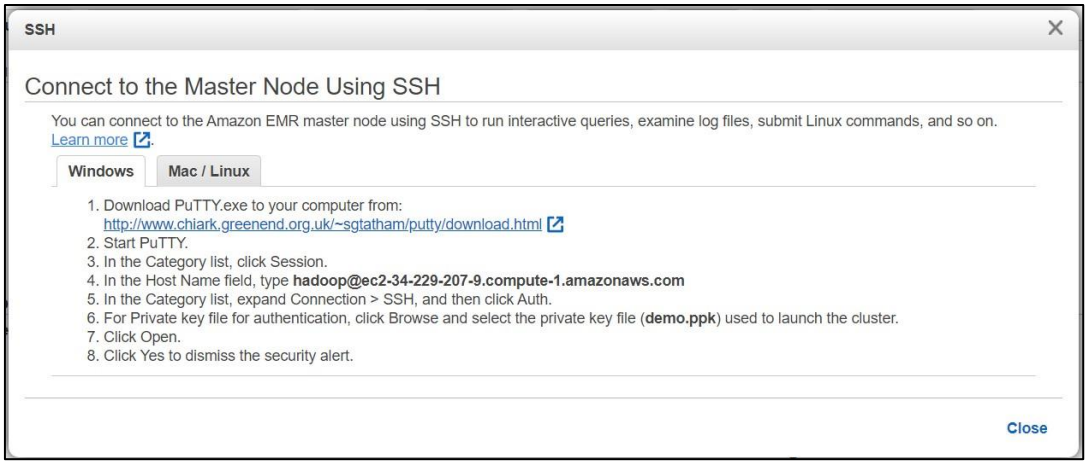
- To proceed first we need create an EMR cluster. So, here we have launched an EMR cluster and it is starting as you can see it is in starting state.



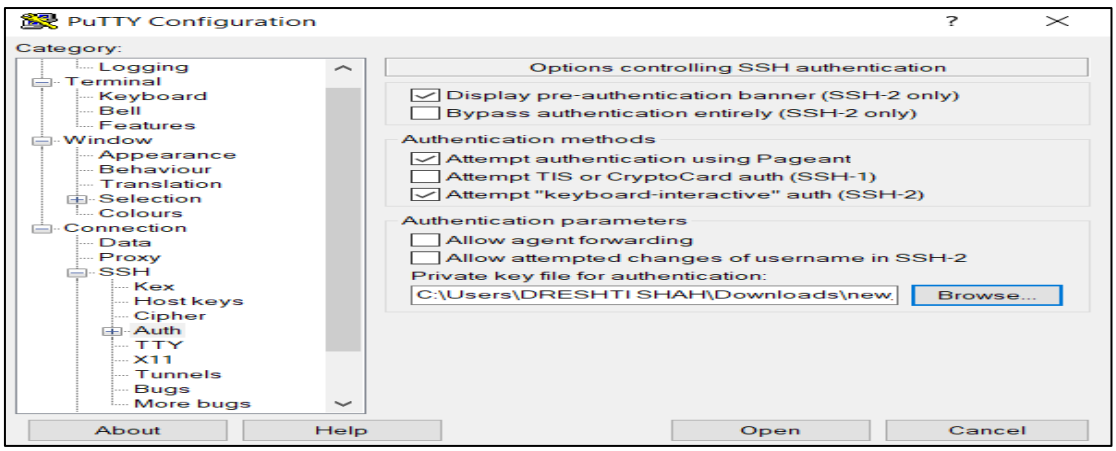
- Now the cluster has started and is in running state so now we can proceed further.



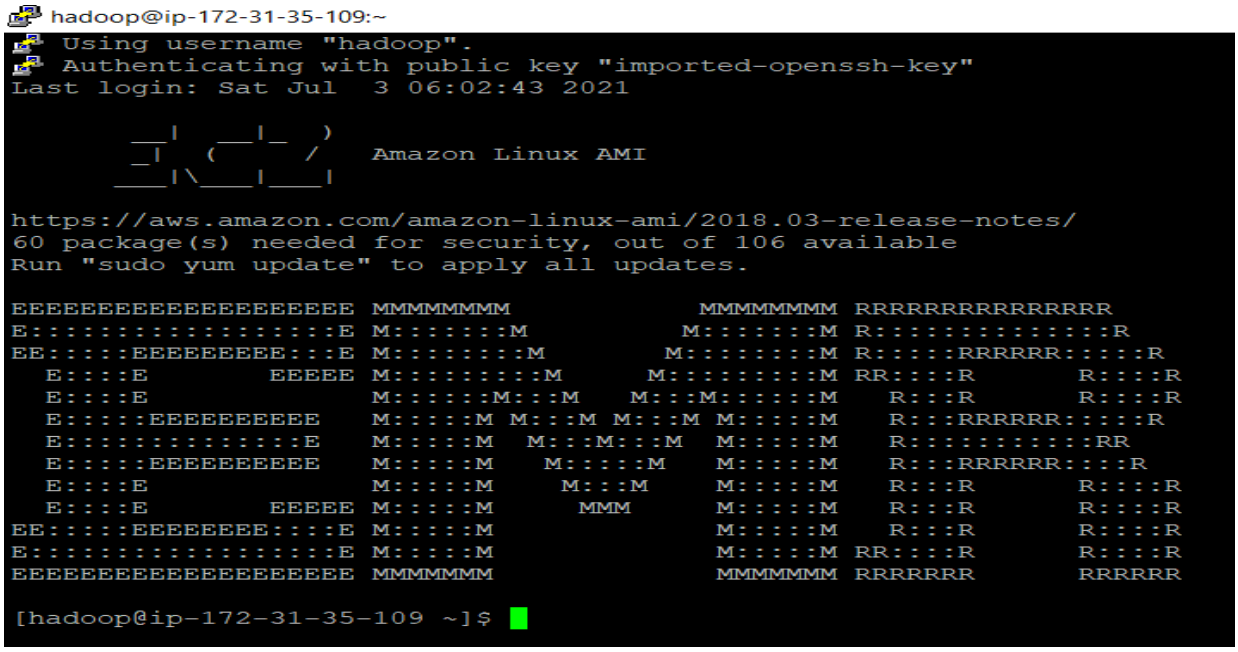
- Proceeding further we will now connect to our master node using ssh. So, we can perform operation on the case.



- Here we are making the connection using putty, as you can see, we have pasted the cluster hostname and also browsed the required key-pairs.



- So, here we have successfully established a connection as we can see the EMR log appear in our putty terminal.



- We were provided with these two links given below which contains the data required for analysis.

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

- Proceeding further, since the data was stored in s3 bucket, we will go ahead and load the data into our HDFS using the Hadoop distcp command as shown below.

```
[hadoop@ip-172-31-35-109 ~]$ hadoop distcp s3://e-commerce-events-ml/2019-Oct.csv /tmp/studydirectory/October
```

```
[hadoop@ip-172-31-35-109 ~]$ hadoop distcp s3://e-commerce-events-ml/2019-Nov.csv /tmp/studydirectory/November
```

- So, as we can see our data is successfully loaded in the directory we had created. Also, we can see that both the files have been loaded in the same directory.

```
[hadoop@ip-172-31-35-109 ~]$ hadoop fs -ls /tmp/studydirectory/
Found 2 items
-rw-r--r--   1 hadoop hadoop  545839412 2021-07-03 06:12 /tmp/studydirectory/November
-rw-r--r--   1 hadoop hadoop  482542278 2021-07-03 06:11 /tmp/studydirectory/October
[hadoop@ip-172-31-35-109 ~]$
```

- Let us proceed and begin with our analysis. So, we enter into hive to perform querying on data as shown below. Also, we created a data base and we will perform the analysis in this data base.

```
[hadoop@ip-172-31-35-109 ~]$ hive

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> create database if not exists ecommerce_casestudy;
OK
Time taken: 0.758 seconds
hive> use ecommerce_casestudy;
OK
Time taken: 0.043 seconds
```

- Here we have created a main table called store in which we have loaded both the files in the same table as both the columns and datatypes are same. Also, we have displayed 10 rows to see if the data is loaded correctly or not as shown below.

```
hive> create table if not exists store(event_time timestamp, event_type string, product_id string, category_id string, category_code string, brand string, price decimal(10,3), user_id strin
g, user_session string) ROW FORMAT SERDE'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH SERDEPROPERTIES ("separatorChar"=",", "quoteChar"="\"", "escapeChar"="\\") stored as textfile LOCAT
ION'/tmp/studydirectory/' TBLPROPERTIES("skip.header.line.count"="1");
OK
Time taken: 0.073 seconds
hive> SELECT * FROM store LIMIT 7;
OK
store.event_time        store.event_type        store.product_id        store.category_id       store.category_code     store.brand     store.price     store.user_id   store.user_session
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     22.22   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
2019-11-01 00:00:24 UTC remove_from_cart        5826182 1487580007483048900             3.33    553329724       2067216c-31b5-455d-a1cc-af0575a34ffb
2019-11-01 00:00:25 UTC view    5856189 1487580009026551821     runail  15.71   562076640       09fafd6c-6c99-46b1-834f-33527f4de241
Time taken: 0.18 seconds, Fetched: 7 row(s)
```

- Describing the table

```
hive> describe store;
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
Time taken: 0.052 seconds, Fetched: 9 row(s)
```

- To optimise we have created a table store_partitioned. This table contains the partitioned version of the main table store. Here we have partitioned on date column.

```
hive> create table if not exists store_partitioned(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)
    > partitioned by (event_date date);
OK
Time taken: 0.152 seconds
hive> set hive.exec.dynamic.partition=true;
hive> set hive.exec.dynamic.partition.mode=nonstrict;
```

- Here we are inserting the data into the partitioned table we had created, basically we coppying data from main table as shown below and we can see it has successfully created the partitions.

```
hive> insert into store_partitioned partition(event_date)
    > select *, to_date(event_time) from store;
Query ID = hadoop_20210703070301_295c714d-1697-4fa3-b60e-a35a979ba383
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0007)

----------------------------------------------------------------------------------------------
        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: 188.46 s
----------------------------------------------------------------------------------------------
Loading data to table ecommerce_casestudy.store_partitioned partition (event_date=null)

Loaded : 61/61 partitions.
        Time taken to load dynamic partitions: 4.82 seconds
        Time taken for adding to write entity : 0.024 seconds
OK
_col0   _col1   _col2   _col3   _col4   _col5   _col6   _col7   _col8   _col9
Time taken: 198.21 seconds
```

- Describing the table

```
hive> describe store_partitioned;
OK
col_name        data_type       comment
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
event_date              date

# Partition Information
# col_name              data_type               comment

event_date              date
Time taken: 0.134 seconds, Fetched: 15 row(s)
```

- Displaying 7 rows of the table

```
hive> SELECT * FROM store_partitioned LIMIT 7;
OK
store_partitioned.event_time    store_partitioned.event_type    store_partitioned.product_id    store_partitioned.category_id   store_partitioned.category_code store_partitioned.brand store
_partitioned.price      store_partitioned.user_id       store_partitioned.user_session  store_partitioned.event_date
NULL    cart    5773203 1487580005134238553             runail  2.620   463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885    2019-10-01
NULL    cart    5773353 1487580005134238553             runail  2.620   463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885    2019-10-01
NULL    cart    5881589 2151191071051219817             lovely  13.480  429681830       49e8d843-adf3-428b-a2c3-fe8bc6a307c9    2019-10-01
NULL    cart    5723490 1487580005134238553             runail  2.620   463240011       26dd6e6e-4dac-4778-8d2c-92e149dab885    2019-10-01
NULL    cart    5881449 1487580013522845895             lovely  0.560   429681830       49e8d843-adf3-428b-a2c3-fe8bc6a307c9    2019-10-01
NULL    cart    5857269 1487580005134238553             runail  2.620   430174032       73dea1e7-664e-43f4-8b30-d32b9d5af04f    2019-10-01
NULL    cart    5739055 1487580008246412266             kapous  4.750   377667011       81326ac6-daa4-4f0a-b488-fd0956a78733    2019-10-01
Time taken: 0.432 seconds, Fetched: 7 row(s)
```

- Now we will further optimise and create buckets on user id. As shown below we have successfully further clustered the data into 27 buckets and copied the data form the partitioned table.

```
hive> set hive.enforce.bucketing=true;
hive> create table if not exists store_bucket (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,
    > event_date date)
    > clustered by (user_id) into 27 buckets;
OK
Time taken: 0.094 seconds
hive> insert into store_bucket
    > select * from store_partitioned;
Query ID = hadoop_20210703072607_3d7bfe58-0d0f-45ca-9bb3-e091a91b212a
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_1625291994888_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED     27         27        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 99.00 s
--------------------------------------------------------------------------------
Loading data to table ecommerce_casestudy.store_bucket
OK
store_partitioned.event_time    store_partitioned.event_type    store_partitioned.product_id    store_partitioned.category_id   store_partitioned.category_code store_partitioned.brand store
_partitioned.price      store_partitioned.user_id       store_partitioned.user_session  store_partitioned.event_date
Time taken: 108.703 seconds
```

- Describing the table

```
hive> describe store_bucket;
OK
col_name            data_type           comment
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
event_date                  date
Time taken: 0.029 seconds, Fetched: 10 row(s)
```

- Displaying 7 rows of the table

```
hive> SELECT * FROM store_bucket LIMIT 7;
OK
store_bucket.event_time store_bucket.event_type store_bucket.product_id store_bucket.category_id       store_bucket.category_code      store_bucket.brand      store_bucket.price      store
_bucket.user_id store_bucket.user_session       store_bucket.event_date
NULL    remove_from_cart        5657910 1487580008145748965                 4.440   562991067       fcd5c51f-87d3-49e1-b5a3-717d0095936e    2019-10-23
NULL    remove_from_cart        5657910 1487580008145748965                 4.440   562991067       fcd5c51f-87d3-49e1-b5a3-717d0095936e    2019-10-23
NULL    view    5649207 1487580013581566154             concept 2.940   563363532       038363d3-5c74-4e91-88e9-a1da43f13c4a    2019-10-23
NULL    remove_from_cart        5705000 1487580008145748965             irisk   1.110   562991067       fcd5c51f-87d3-49e1-b5a3-717d0095936e    2019-10-23
NULL    view    5817161 1487580008800059394             masura  5.540   524435310       793791d9-1d89-41b4-83f5-4ed92f0a0e87    2019-10-23
NULL    cart    5711127 1487580005008409427             f.o.x   4.830   563295627       29965673-bad3-4f03-ad2b-4a6be22d54d4    2019-10-23
NULL    remove_from_cart        5688826 1487580008145748965                 4.840   562991067       fcd5c51f-87d3-49e1-b5a3-717d0095936e    2019-10-23
Time taken: 0.149 seconds, Fetched: 7 row(s)
```

- Now we are done with all the table creation, data loading, partitioning, and bucketing. Let's proceed further with our analysis. We have been given 8 questions to solve. So, let's go ahead and solve them.

VALUABLE INSIGHTS

Q1: Find the total revenue generated due to purchases made in October?

- We have the total revenue which came out to be 1211538.430 $ for the month of October.

```
hive> SELECT sum(price) as revenue_generated from store_bucket where event_type='purchase' and month(event_date)=10;
Query ID = hadoop_20210703073125_ecefb48-59b5-41cd-b272-99754530f714
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 26.04 s
--------------------------------------------------------------------------------
OK
revenue_generated
1211538.430
Time taken: 27.079 seconds, Fetched: 1 row(s)
```

Q2: Write a query to yield the total sum of purchases per month in a single output.

- We have found the total no. of products as well as the revenue generated for both the months. So, we found out that 245624 products sold and revenue is 1211538 for the month of October, 322417 products sold and revenue is 1531016 for the month of November.

```
hive> select count(product_id) as products_sold, sum(price) as revenue_generated ,month(event_date) as month from store_bucket where event_type='purchase' group by (month(event_date));
Query ID = hadoop_20210703073311_2bd260e3-1c00-41c8-aff8-867a3becd2d2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0008)

--------------------------------------------------------------------------------
        VERTICES      MODE       STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 26.72 s
--------------------------------------------------------------------------------
OK
products_sold   revenue_generated       month
245624  1211538.430     10
322417  1531016.900     11
Time taken: 27.384 seconds, Fetched: 2 row(s)
```

Q3: Write a query to find the change in revenue generated due to purchases from October to November?

- Here we are trying to find how much change was observed in the revenue generated from October to November. We can see an increase of 319478 $

```
hive> WITH revenue_change as
    > (
    > SELECT sum(case when month(event_date)=10 then price else 0 end) as Oct_Sales,
    >                sum(case when month(event_date)=11 then price else 0 end) as Nov_Sales
    >                 from store_bucket
    >                 where event_type='purchase'
    > )
    > SELECT  Oct_Sales, Nov_sales, Nov_Sales - Oct_Sales as CHANGE_IN_REVENUE from revenue_change;
Query ID = hadoop_20210704103351_8c0f1238-8959-4df2-8502-ac3796afd4ed
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_1625391434489_0005)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 28.44 s
--------------------------------------------------------------------------------
OK
oct_sales       nov_sales       change_in_revenue
1211538.430     1531016.900     319478.470
Time taken: 37.737 seconds, Fetched: 1 row(s)
```

Q4: Find distinct categories of products. Categories with null category code can be ignored?

- We are just finding the unique categories of products here.

```
hive> select distinct category_code from store_bucket where category_code  != 'null' and category_code != '';
Query ID = hadoop_20210703075511_00a6ba7b-b1b6-40b7-b1bb-973e47e55859
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0009)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      4          4        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 26.40 s
--------------------------------------------------------------------------------
OK
category_code
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: 27.162 seconds, Fetched: 11 row(s)
```

Q5: Find the total number of products available under each category?

- Here we are finding the how many products are there under each category

```
hive> select category_code as categories, count(distinct product_id) as number_of_products from store_bucket where category_code !='null'  and category_code !='' group by category_code;
Query ID = hadoop_20210703075745_2ecbccaf-ce84-40ba-9e16-cae042428a66
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0009)

--------------------------------------------------------------------------------
       VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      4          4        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 26.43 s
--------------------------------------------------------------------------------
OK
categories      number_of_products
apparel.glove   78
appliances.environment.air_conditioner  26
appliances.environment.vacuum   85
furniture.living_room.chair     2
accessories.bag 42
accessories.cosmetic_bag        16
appliances.personal.hair_cutter 9
furniture.bathroom.bath 55
furniture.living_room.cabinet   6
sport.diving    1
stationery.cartrige     138
Time taken: 27.032 seconds, Fetched: 11 row(s)
```

Q6: Which brand had the maximum sales in October and November combined?

- So, we can see here the brand Runail had maximum sales in both the month October and November combined i.e., a total of 149297 $ revenue generated.

```
hive> select brand, sum(price) as total_sales from store_bucket where event_type='purchase' and brand !='null' and brand !='' group by brand order by total_sales desc limit 1;
Query ID = hadoop_20210703080017_e3e3bb0f-8a2f-4cff-a8df-bf310ad9365c
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0009)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 25.10 s
--------------------------------------------------------------------------------
OK
brand   total_sales
runail  148297.940
Time taken: 25.758 seconds, Fetched: 1 row(s)
```

Q7: Which brands increased their sales from October to November?

- Here is the list of brands that increased their sales from October to November i.e., a total of 160 brands have increased the revenue.

```
hive> with total_sales as ( select brand, sum(case when month(event_date)=10 then price else 0 end) as oct_sales_count, sum(case when month(event_date)=11 then price else 0 end) as nov_sale
s_count from store_bucket where event_type='purchase' and brand !='null' and brand !='' group by brand) select brand from total_sales where nov_sales_count>oct_sales_count;
Query ID = hadoop_20210703080510_0628a653-385b-4727-8bad-f8c70cfcb870
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625291994888_0009)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 27.80 s
--------------------------------------------------------------------------------
OK
brand
art-visage
artex
barbie
batiste
beautix
beautyblender
bioaqua
biore
blixz
browxenna
carmex
concept
cutrin
deoproce
domix
ecocraft
ecolab
egomania
ellips
elskin
entity
eos
f.o.x
farmavita
fedua
fly
freshbubble
gehwol
glysolid
greymy
happyfons
haruyama
```

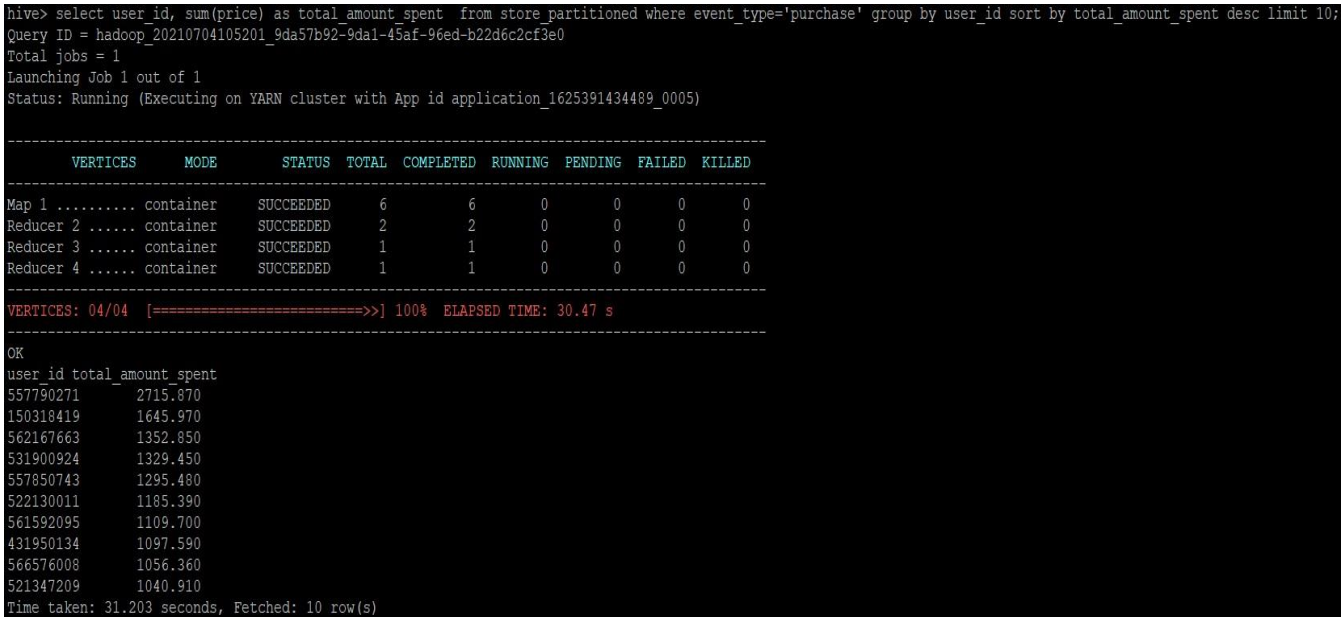
```
helloganic
inm
insight
jaguar
joico
juno
kaaral
kamill
kares
kaypro
keen
konad
laboratorium
levissime
lianail
likato
limoni
lovely
mane
marathon
markell
masura
mavala
milv
miskin
missha
moyou
nagaraku
naomi
nefertiti
nirvel
nitrile
orly
osmo
ovale
plazan
profhenna
protokeratin
provoc
runail
shik
skinlite
smart
soleo
sophin
strong
trind
uno
uskusi
yoko
airnails
aura
balbcare
beauty-free
beauugreen
benovy
binacil
bluesky
bodyton
bpw.style
candy
chi
coifin
cosima
cosmoprofi
cristalinas
de.lux
depilflax
dewal
dizao
elizavecca
enjoy
estel
estelare
farmona
finish
foamie
freedecor
godefroy
grace
grattol
igrobeauty
ingarden
irisk
italwax
jas
jessnail
kapous
kerasys
kims
kinetics
kiss
kocostar
koelcia
koelf
kosmekka
lador
ladykin
latinoil
levrana
lowence
marutaka-foot
matreshka
matrix
metzger
neoleor
oniq
polarus
profepil
rasyan
refectocil
rosi
roubloff
s.care
sanoto
severina
shary
skinity
solomeya
staleks
supertan
swarovski
tertio
treaclemoon
veraclara
vilenta
yu-r
zeitun
Time taken: 28.369 seconds, Fetched: 160 row(s)
```

Q8: 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?

- So, here is the list of top 10 users who have spent the most.

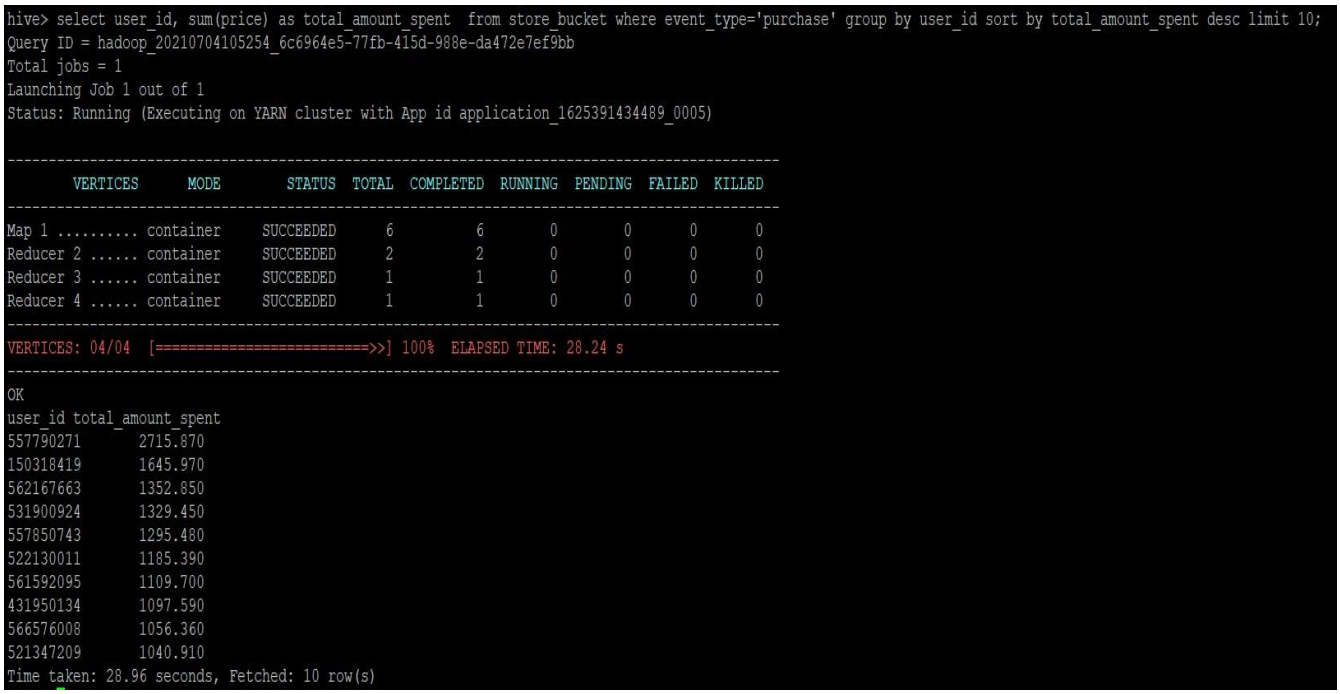
```
hive> select user_id, sum(price) as total_amount_spent  from store where event_type='purchase' group by user_id sort by total_amount_spent desc limit 10;
Query ID = hadoop_20210704105039_05de7ad5-0a3a-4b80-a2b3-512c8fa5d1c8
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625391434489_0005)

----------------------------------------------------------------------------------------------
        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
Reducer 3 ...... container     SUCCEEDED      2          2        0        0       0       0
Reducer 4 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 04/04  [==========================>>] 100%  ELAPSED TIME: 57.94 s
----------------------------------------------------------------------------------------------
OK
user_id total_amount_spent
557790271       2715.869999999991
150318419       1645.97
562167663       1352.8500000000004
531900924       1329.4500000000003
557850743       1295.4800000000002
522130011       1185.3899999999994
561592095       1109.6999999999996
431950134       1097.5899999999995
566576008       1056.3600000000017
521347209       1040.9099999999999
Time taken: 58.555 seconds, Fetched: 10 row(s)
```

- Now, if you notice here the execution time is 58 seconds. Which is too high, let's optimise and reduce this.
- So as you can see below we executed the same query on the partitioned table (store_partitioned) which we had created earlier. So, the query execution time has reduce to 31 seconds which is a drastic change.



```
hive> select user_id, sum(price) as total_amount_spent  from store_partitioned where event_type='purchase' group by user_id sort by total_amount_spent desc limit 10;
Query ID = hadoop_20210704105201_9da57b92-9da1-45af-96ed-b22d6c2cf3e0
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625391434489_0005)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 4 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 04/04  [==========================>>] 100%  ELAPSED TIME: 30.47 s
--------------------------------------------------------------------------------
OK
user_id total_amount_spent
557790271       2715.870
150318419       1645.970
562167663       1352.850
531900924       1329.450
557850743       1295.480
522130011       1185.390
561592095       1109.700
431950134       1097.590
566576008       1056.360
521347209       1040.910
Time taken: 31.203 seconds, Fetched: 10 row(s)
```

- We further tried to optimise it by executing the same query but on the bucketed table (store_bucket) we had created. So, you can see the execution time has further reduced to 28 seconds. Not much of a change but still the job has been optimised as much as possible.



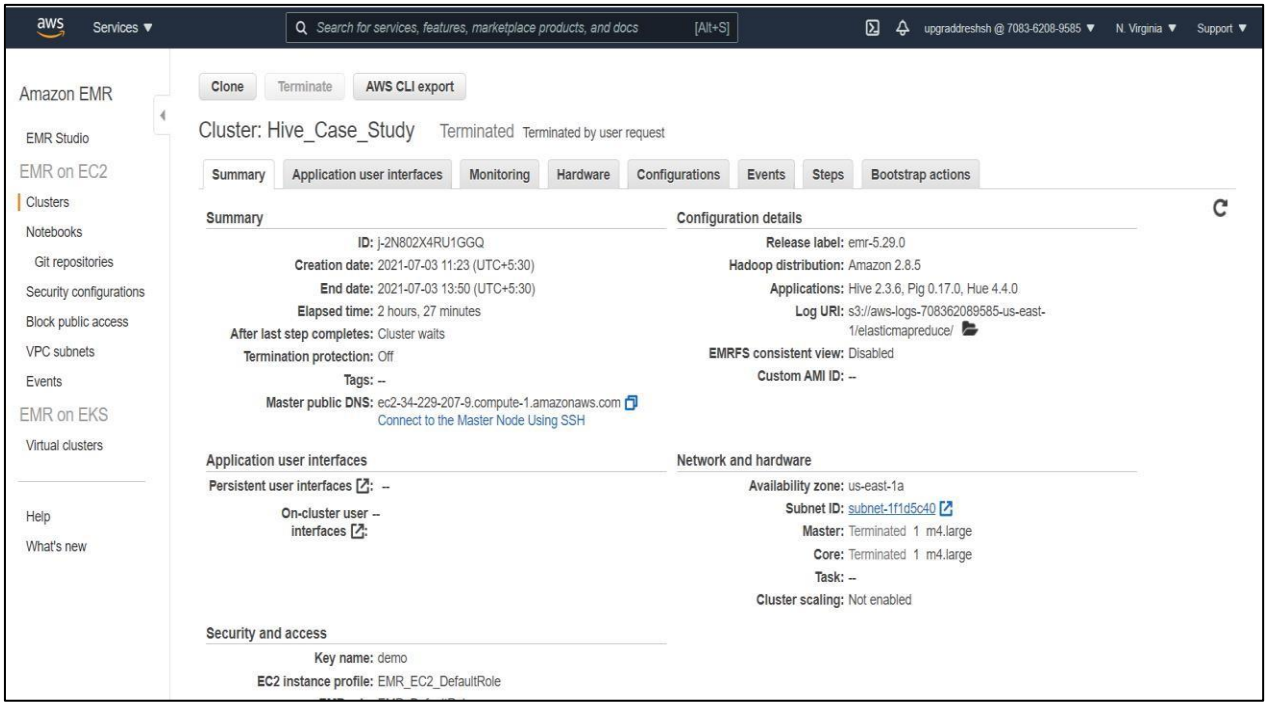
```
hive> select user_id, sum(price) as total_amount_spent  from store_bucket where event_type='purchase' group by user_id sort by total_amount_spent desc limit 10;
Query ID = hadoop_20210704105254_6c6964e5-77fb-415d-988e-da472e7ef9bb
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1625391434489_0005)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 4 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 04/04  [==========================>>] 100%  ELAPSED TIME: 28.24 s
--------------------------------------------------------------------------------
OK
user_id total_amount_spent
557790271       2715.870
150318419       1645.970
562167663       1352.850
531900924       1329.450
557850743       1295.480
522130011       1185.390
561592095       1109.700
431950134       1097.590
566576008       1056.360
521347209       1040.910
Time taken: 28.96 seconds, Fetched: 10 row(s)
```

- Now since all our work is done and the analysis is complete let's go ahead and drop the database and data associated to it.

```
hive> DROP DATABASE if exists ecommerce_casestudy cascade;
OK
Time taken: 0.87 seconds
hive> show databases;
OK
database_name
default
Time taken: 0.024 seconds, Fetched: 1 row(s)
hive> exit;
[hadoop@ip-172-31-35-109 ~]$ hadoop fs -ls /tmp/studydirectory/
ls: `/tmp/studydirectory/': No such file or directory
[hadoop@ip-172-31-35-109 ~]$ hadoop fs -ls /tmp/
Found 2 items
drwxrwxrwx   - mapred mapred          0 2021-07-03 05:58 /tmp/hadoop-yarn
drwx-wx-wx   - hive   hadoop          0 2021-07-03 06:16 /tmp/hive
[hadoop@ip-172-31-35-109 ~]$
```

- So, as you can see above the data base was dropped. Also, we checked the HDFS the files have also been deleted and the directory as well.

- So now let's go ahead and terminate the cluster as our analysis is complete.



- As you can see the cluster has been terminated.

THE END