Case Study Submitted by: Roshan Chandru & Gunjan Bhardwaj

HIVE CASE STUDY

Problem Statement:

With online sales gaining popularity, tech companies are exploring ways to improve their sales by analyzing customer behavior 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.

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

The data is available from the link provided:

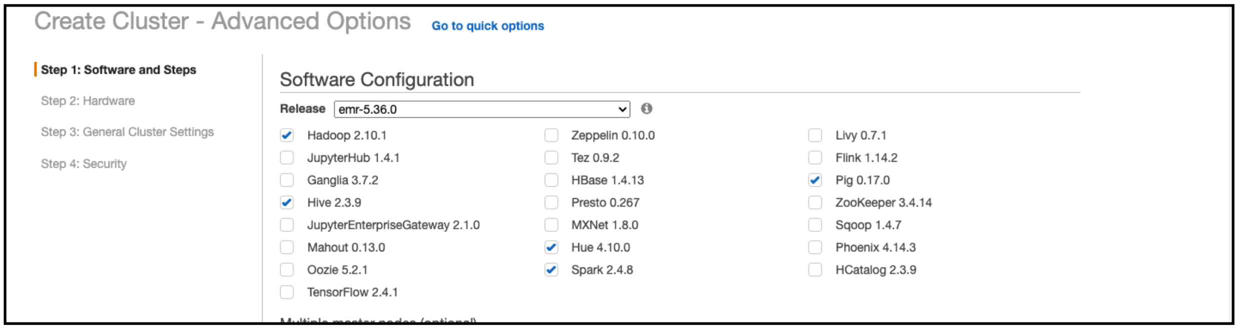
https://e-commerce-events-ml.s3.amazonaws.com/2019-Oct.csv
https://e-commerce-events-ml.s3.amazonaws.com/2019-Nov.csv

Overview of steps:

- **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

Data Collection and Processing:

EMR Cluster Creation



Create Cluster - Advanced Options   Go to quick options

Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security

Software Configuration

Release emr-5.36.0

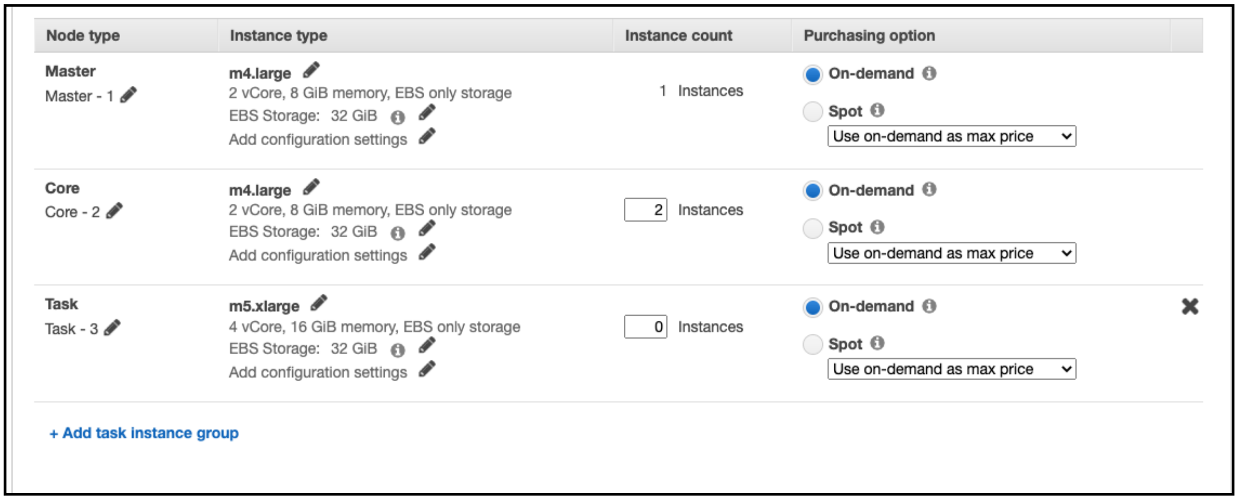
☑ Hadoop 2.10.1
☐ JupyterHub 1.4.1
☐ Ganglia 3.7.2
☑ Hive 2.3.9
☐ JupyterEnterpriseGateway 2.1.0
☐ Mahout 0.13.0
☐ Oozie 5.2.1
☐ TensorFlow 2.4.1

☐ Zeppelin 0.10.0
☐ Tez 0.9.2
☐ HBase 1.4.13
☐ Presto 0.267
☐ MXNet 1.8.0
☑ Hue 4.10.0
☑ Spark 2.4.8

☐ Livy 0.7.1
☐ Flink 1.14.2
☑ Pig 0.17.0
☐ ZooKeeper 3.4.14
☐ Sqoop 1.4.7
☐ Phoenix 4.14.3
☐ HCatalog 2.3.9

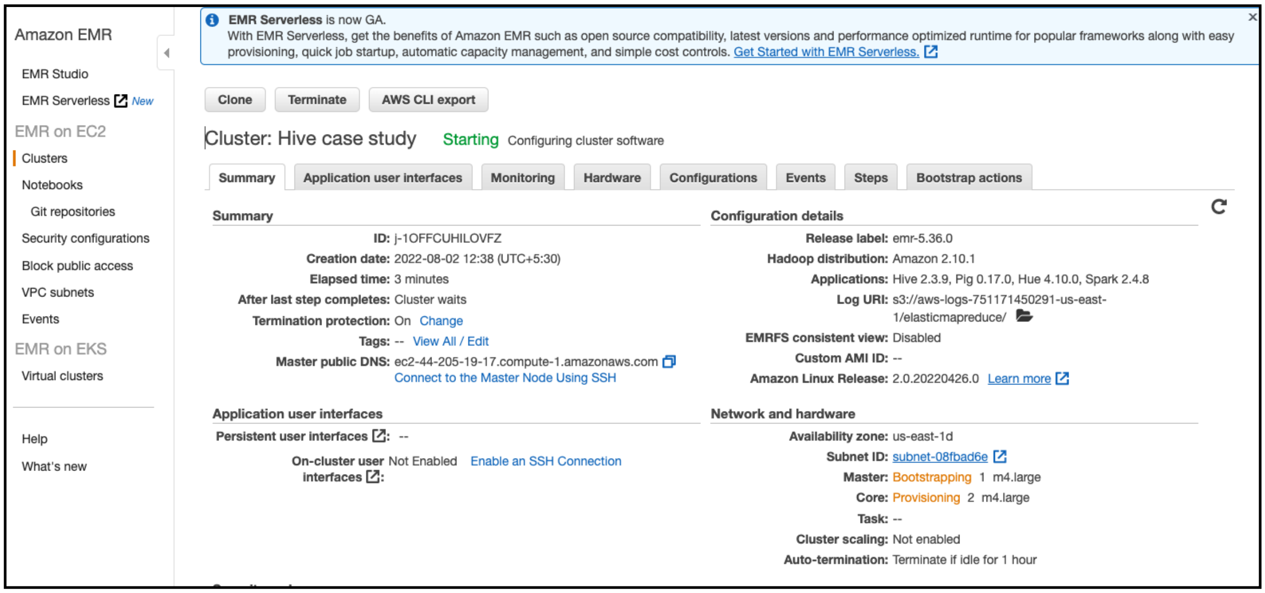
Multiple master nodes (optional)

Hardware Configuration Page > To define the cluster & nodes : Instance type for both master &core nodes are M4.large



Node type	Instance type	Instance count	Purchasing option
Master Master - 1	**m4.large** 2 vCore, 8 GiB memory, EBS only storage EBS Storage: 32 GiB Add configuration settings	1 Instances	● **On-demand** ○ **Spot** Use on-demand as max price
Core Core - 2	**m4.large** 2 vCore, 8 GiB memory, EBS only storage EBS Storage: 32 GiB Add configuration settings	2 Instances	● **On-demand** ○ **Spot** Use on-demand as max price
Task Task - 3	**m5.xlarge** 4 vCore, 16 GiB memory, EBS only storage EBS Storage: 32 GiB Add configuration settings	0 Instances	● **On-demand** ○ **Spot** Use on-demand as max price

+ Add task instance group



Amazon EMR

EMR Studio
EMR Serverless New
EMR on EC2
Clusters
Notebooks
Git repositories
Security configurations
Block public access
VPC subnets
Events
EMR on EKS
Virtual clusters

Help
What's new

EMR Serverless is now GA.
With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. Get Started with EMR Serverless.

Clone | Terminate | AWS CLI export

Cluster: Hive case study   Starting Configuring cluster software

Summary | Application user interfaces | Monitoring | Hardware | Configurations | Events | Steps | Bootstrap actions

Summary

ID: j-1OFFCUHILOVFZ
Creation date: 2022-08-02 12:38 (UTC+5:30)
Elapsed time: 3 minutes
After last step completes: Cluster waits
Termination protection: On Change
Tags: -- View All / Edit
Master public DNS: ec2-44-205-19-17.compute-1.amazonaws.com
Connect to the Master Node Using SSH

Application user interfaces

Persistent user interfaces: --
On-cluster user interfaces: Not Enabled Enable an SSH Connection

Configuration details

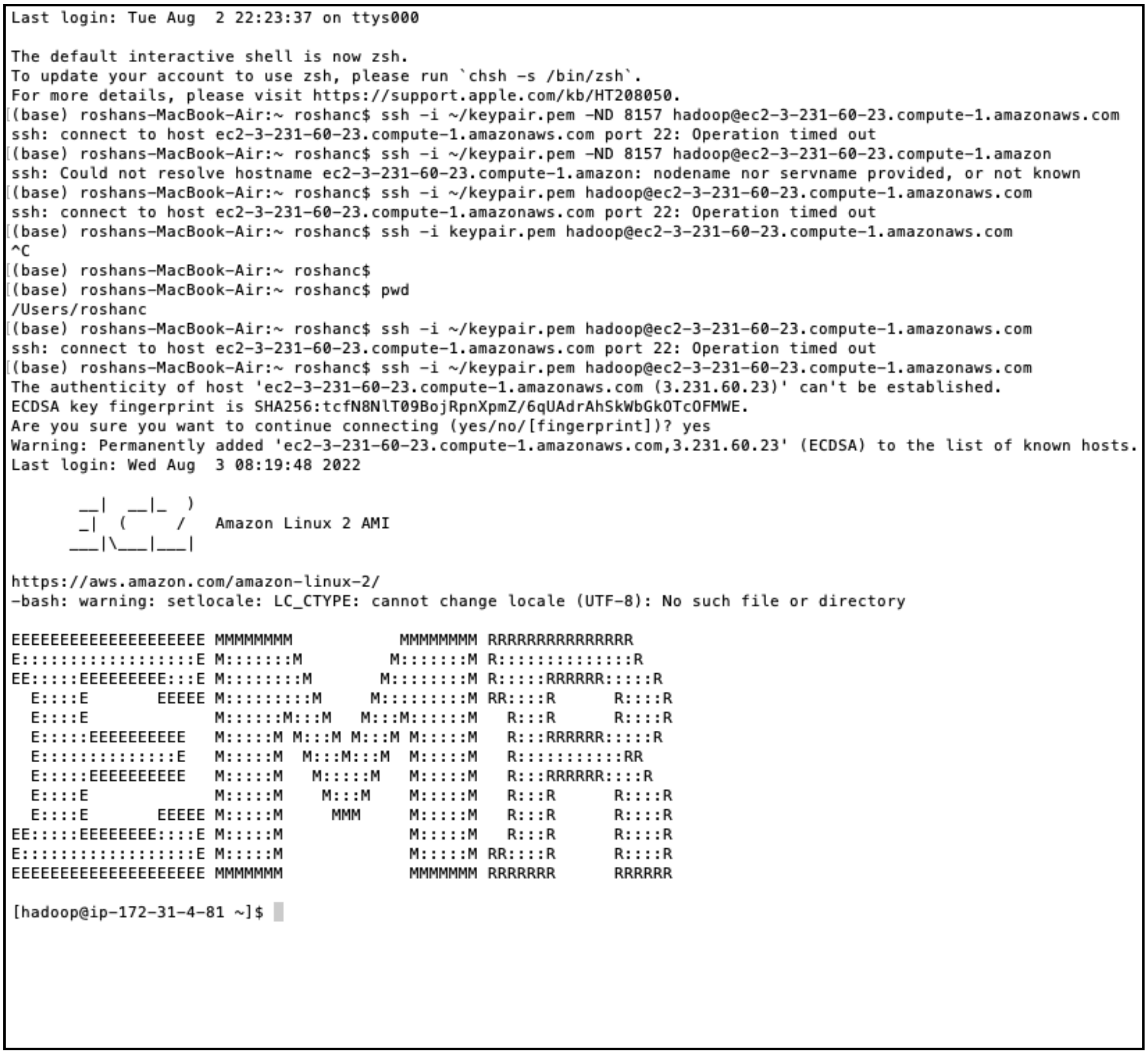
Release label: emr-5.36.0
Hadoop distribution: Amazon 2.10.1
Applications: Hive 2.3.9, Pig 0.17.0, Hue 4.10.0, Spark 2.4.8
Log URI: s3://aws-logs-751171450291-us-east-1/elasticmapreduce/
EMRFS consistent view: Disabled
Custom AMI ID: --
Amazon Linux Release: 2.0.20220426.0 Learn more

Network and hardware

Availability zone: us-east-1d
Subnet ID: subnet-08fbad6e
Master: Bootstrapping 1 m4.large
Core: Provisioning 2 m4.large
Task: --
Cluster scaling: Not enabled
Auto-termination: Terminate if idle for 1 hour

Hadoop & Hive Queries :

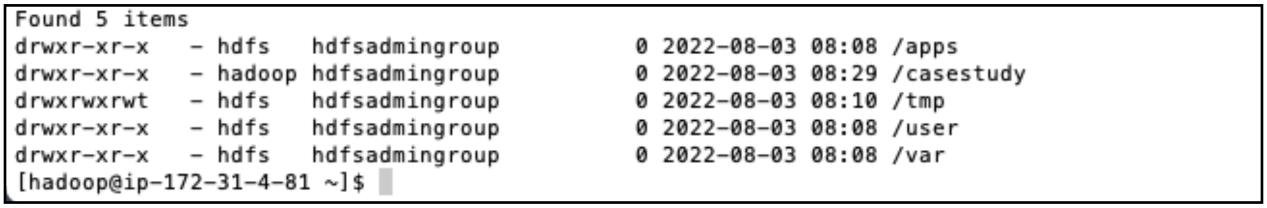
Terminal > Connecting to EMR Cluster using ssh.



```
Last login: Tue Aug  2 22:23:37 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i ~/keypair.pem -ND 8157 hadoop@ec2-3-231-60-23.compute-1.amazonaws.com
ssh: connect to host ec2-3-231-60-23.compute-1.amazonaws.com port 22: Operation timed out
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i ~/keypair.pem -ND 8157 hadoop@ec2-3-231-60-23.compute-1.amazon
ssh: Could not resolve hostname ec2-3-231-60-23.compute-1.amazon: nodename nor servname provided, or not known
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i ~/keypair.pem hadoop@ec2-3-231-60-23.compute-1.amazonaws.com
ssh: connect to host ec2-3-231-60-23.compute-1.amazonaws.com port 22: Operation timed out
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i keypair.pem hadoop@ec2-3-231-60-23.compute-1.amazonaws.com
^C
[(base) roshans-MacBook-Air:~ roshanc$
[(base) roshans-MacBook-Air:~ roshanc$ pwd
/Users/roshanc
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i ~/keypair.pem hadoop@ec2-3-231-60-23.compute-1.amazonaws.com
ssh: connect to host ec2-3-231-60-23.compute-1.amazonaws.com port 22: Operation timed out
[(base) roshans-MacBook-Air:~ roshanc$ ssh -i ~/keypair.pem hadoop@ec2-3-231-60-23.compute-1.amazonaws.com
The authenticity of host 'ec2-3-231-60-23.compute-1.amazonaws.com (3.231.60.23)' can't be established.
ECDSA key fingerprint is SHA256:tcfN8NlT09BojRpnXpmZ/6qUAdrAhSkWbGkOTcOFMWE.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-231-60-23.compute-1.amazonaws.com,3.231.60.23' (ECDSA) to the list of known hosts.
Last login: Wed Aug  3 08:19:48 2022

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-4-81 ~]$
```

Creating a directory “casestudy”

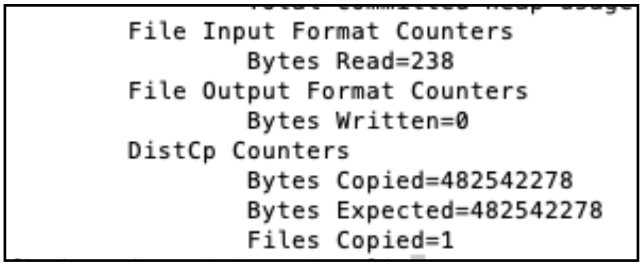
hadoop fs -mkdir /
casestudyhadoop fs -ls /



```
Found 5 items
drwxr-xr-x   - hdfs   hdfsadmingroup          0 2022-08-03 08:08 /apps
drwxr-xr-x   - hadoop hdfsadmingroup          0 2022-08-03 08:29 /casestudy
drwxrwxrwt   - hdfs   hdfsadmingroup          0 2022-08-03 08:10 /tmp
drwxr-xr-x   - hdfs   hdfsadmingroup          0 2022-08-03 08:08 /user
drwxr-xr-x   - hdfs   hdfsadmingroup          0 2022-08-03 08:08 /var
[hadoop@ip-172-31-4-81 ~]$
```

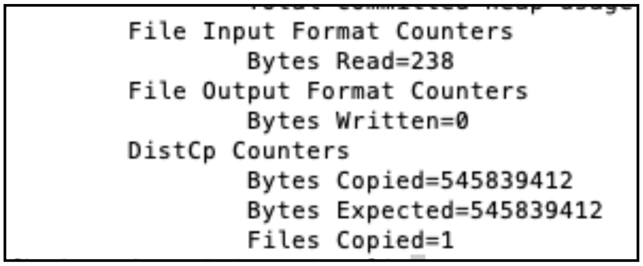
Loading the datasets into HDFS from S3:

hadoop distcp 's3://e-commerce-events-ml/2019-Oct.csv' /casestudy/2019_Oct.csv



```
        File Input Format Counters
                Bytes Read=238
        File Output Format Counters
                Bytes Written=0
        DistCp Counters
                Bytes Copied=482542278
                Bytes Expected=482542278
                Files Copied=1
```

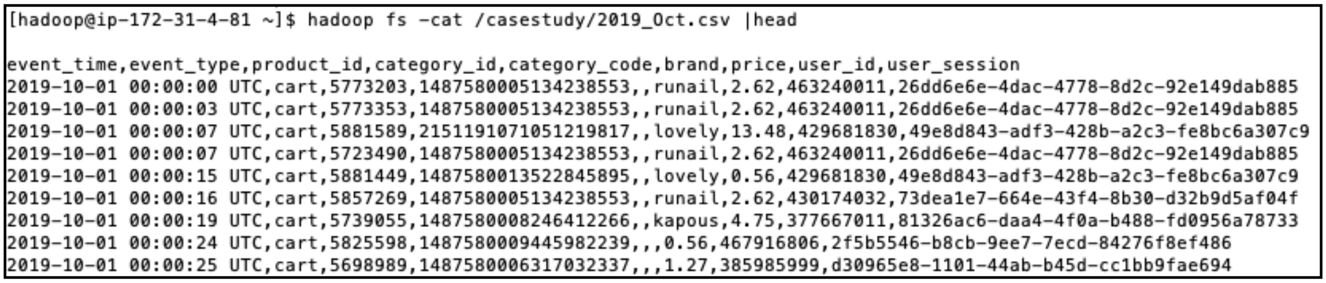
hadoop distcp 's3://e-commerce-events-ml/2019-Nov.csv' /casestudy/2019_Nov.csv



```
        File Input Format Counters
                Bytes Read=238
        File Output Format Counters
                Bytes Written=0
        DistCp Counters
                Bytes Copied=545839412
                Bytes Expected=545839412
                Files Copied=1
```

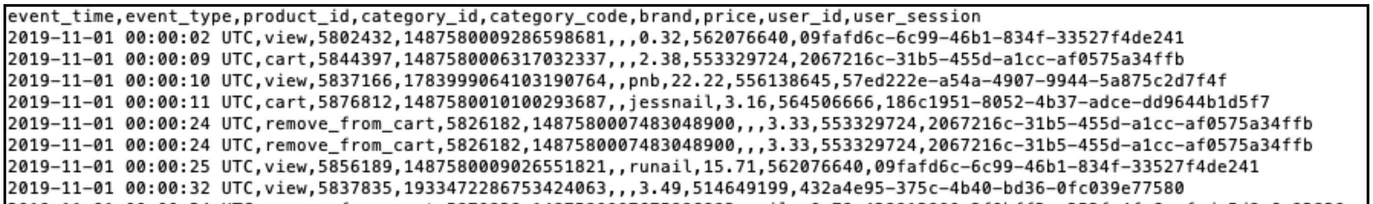
Viewing the data

hadoop fs -cat /casestudy/2019_Oct.csv |head



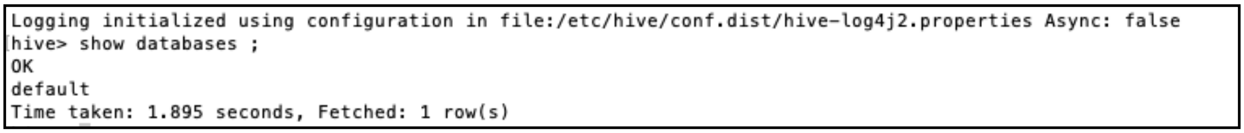
```
[hadoop@ip-172-31-4-81 ~]$ hadoop fs -cat /casestudy/2019_Oct.csv |head

event_time,event_type,product_id,category_id,category_code,brand,price,user_id,user_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,,,0.56,467916806,2f5b5546-b8cb-9ee7-7ecd-84276f8ef486
2019-10-01 00:00:25 UTC,cart,5698989,1487580006317032337,,,1.27,385985999,d30965e8-1101-44ab-b45d-cc1bb9fae694
```

hadoop fs -cat /casestudy/2019_Nov.csv | head



```
event_time,event_type,product_id,category_id,category_code,brand,price,user_id,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
2019-11-01 00:00:32 UTC,view,5837835,1933472286753424063,,,3.49,514649199,432a4e95-375c-4b40-bd36-0fc039e77580
2019-11-01 00:00:34 UTC,remove_from_cart,5870838,1487580007675986893,,milv,0.79,429913900,2f0bff3c-252f-4fe6-afcd-5d8a6a92839a
```

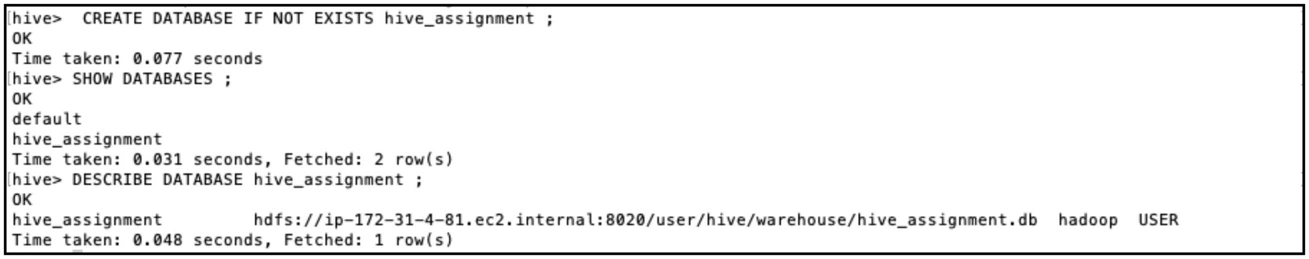
Datasets are successfully loaded.
Launch Hive



```
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> show databases ;
OK
default
Time taken: 1.895 seconds, Fetched: 1 row(s)
```

Creating new database “hive_assignment”

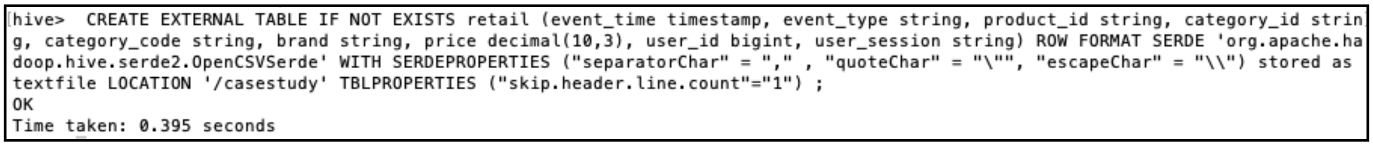
```
hive> CREATE DATABASE IF NOT EXISTS hive_assignment ;
hive> SHOW DATABASES ;
hive> DESCRIBE DATABASE hive_assignment ;
```



```
hive>  CREATE DATABASE IF NOT EXISTS hive_assignment ;
OK
Time taken: 0.077 seconds
hive> SHOW DATABASES ;
OK
default
hive_assignment
Time taken: 0.031 seconds, Fetched: 2 row(s)
hive> DESCRIBE DATABASE hive_assignment ;
OK
hive_assignment         hdfs://ip-172-31-4-81.ec2.internal:8020/user/hive/warehouse/hive_assignment.db  hadoop  USER
Time taken: 0.048 seconds, Fetched: 1 row(s)
```

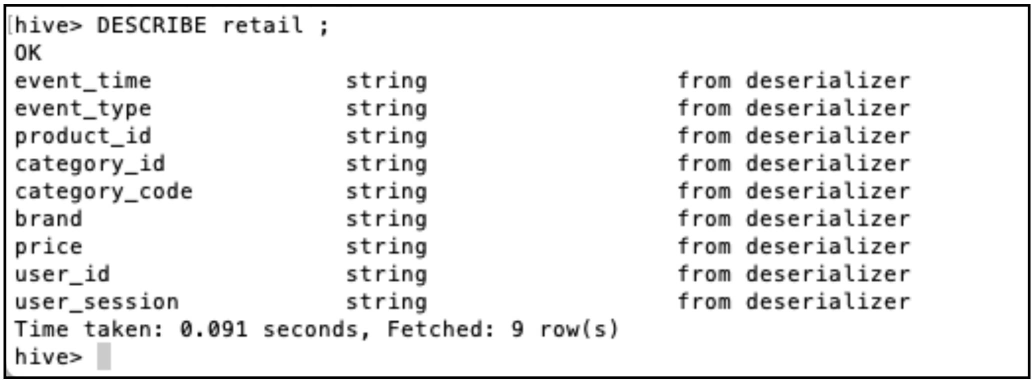
Creating new table “retail”

```
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 '/casestudy' TBLPROPERTIES ("skip.header.line.count"="1") ;
```



```
hive>  CREATE EXTERNAL TABLE IF NOT EXISTS retail (event_time timestamp, event_type string, product_id string, category_id strin
g, category_code string, brand string, price decimal(10,3), user_id bigint, user_session string) ROW FORMAT SERDE 'org.apache.ha
doop.hive.serde2.OpenCSVSerde' WITH SERDEPROPERTIES ("separatorChar" = "," , "quoteChar" = "\"", "escapeChar" = "\\") stored as
textfile LOCATION '/casestudy' TBLPROPERTIES ("skip.header.line.count"="1") ;
OK
Time taken: 0.395 seconds
```

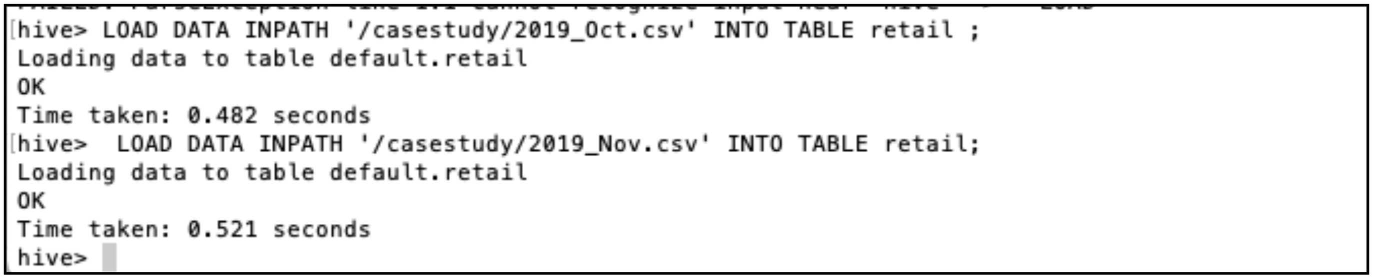
```
hive> DESCRIBE retail ;
```



```
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: 0.091 seconds, Fetched: 9 row(s)
hive>
```

Loading data into table “retail”;

hive> LOAD DATA INPATH '/casestudy/2019_Oct.csv' INTO TABLE retail ;
hive> LOAD DATA INPATH '/casestudy/2019_Nov.csv' INTO TABLE retail;

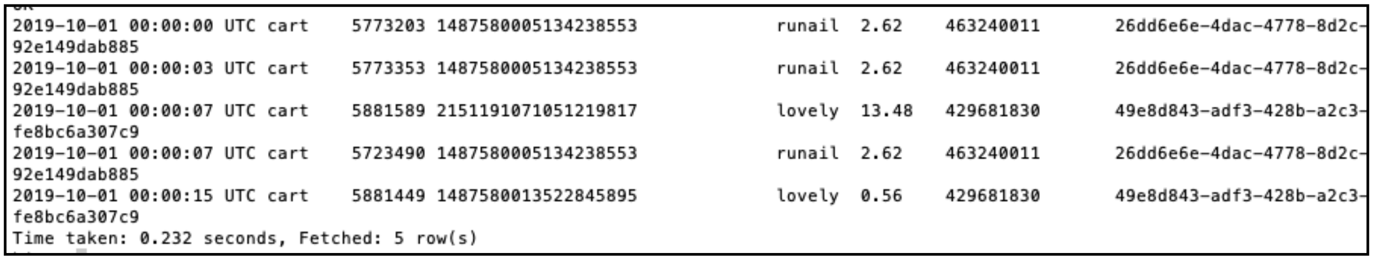


```
hive> LOAD DATA INPATH '/casestudy/2019_Oct.csv' INTO TABLE retail ;
Loading data to table default.retail
OK
Time taken: 0.482 seconds
hive>  LOAD DATA INPATH '/casestudy/2019_Nov.csv' INTO TABLE retail;
Loading data to table default.retail
OK
Time taken: 0.521 seconds
hive>
```

Performing data check:

hive> SELECT * FROM retail WHERE MONTH(event_time)=11 limit 5;

hive> SELECT * FROM retail WHERE MONTH(event_time)=10 limit 5 ;

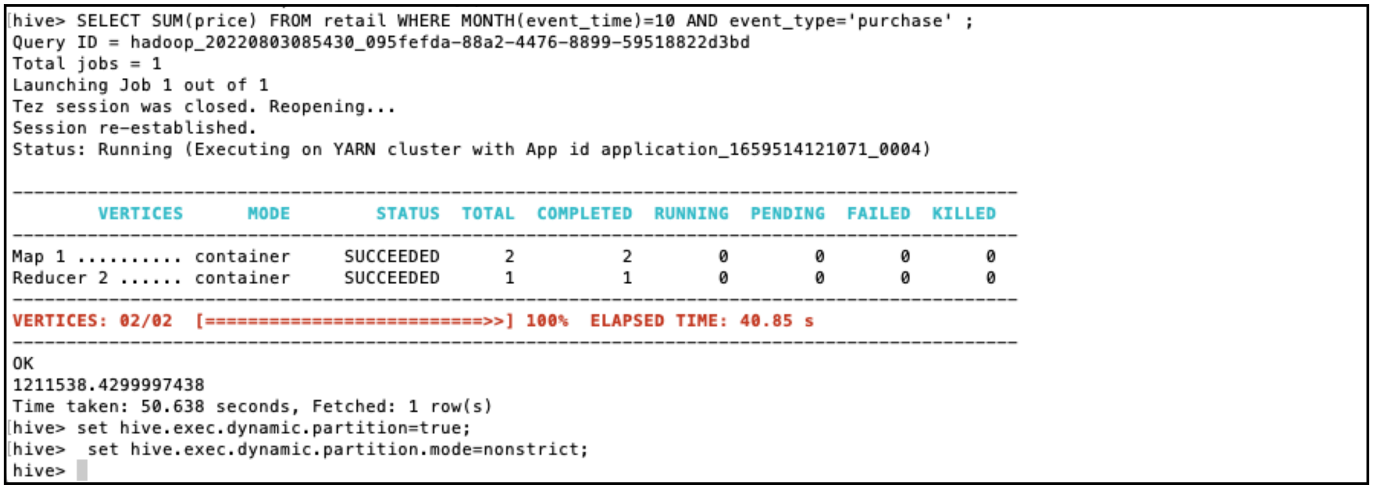


```
2019-10-01 00:00:00 UTC cart    5773203 1487580005134238553            runail  2.62    463240011       26dd6e6e-4dac-4778-8d2c-
92e149dab885
2019-10-01 00:00:03 UTC cart    5773353 1487580005134238553            runail  2.62    463240011       26dd6e6e-4dac-4778-8d2c-
92e149dab885
2019-10-01 00:00:07 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: 0.232 seconds, Fetched: 5 row(s)
```

You are required to provide answers to the questions given below

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

hive> 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_20220803085430_095fefda-88a2-4476-8899-59518822d3bd
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_1659514121071_0004)

----------------------------------------------------------------------------------------------
        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: 40.85 s
----------------------------------------------------------------------------------------------
OK
1211538.4299997438
Time taken: 50.638 seconds, Fetched: 1 row(s)
hive> set hive.exec.dynamic.partition=true;
hive>  set hive.exec.dynamic.partition.mode=nonstrict;
hive>
```

Time taken to execute the above query is 40.85 sec.

DYNAMIC PARTITIONING

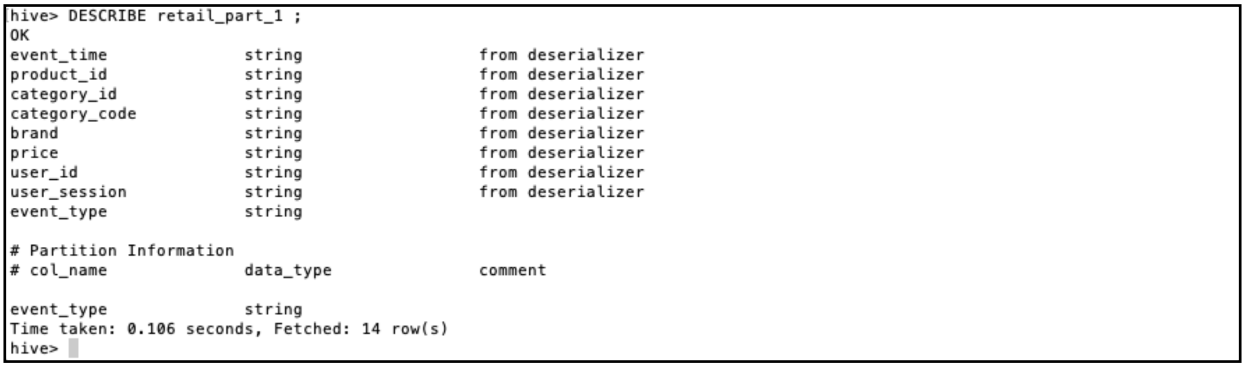
hive> set hive.exec.dynamic.partition=true;
hive> set hive.exec.dynamic.partition.mode=nonstrict;

PARTITION TABLE 1: retail_part_1

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

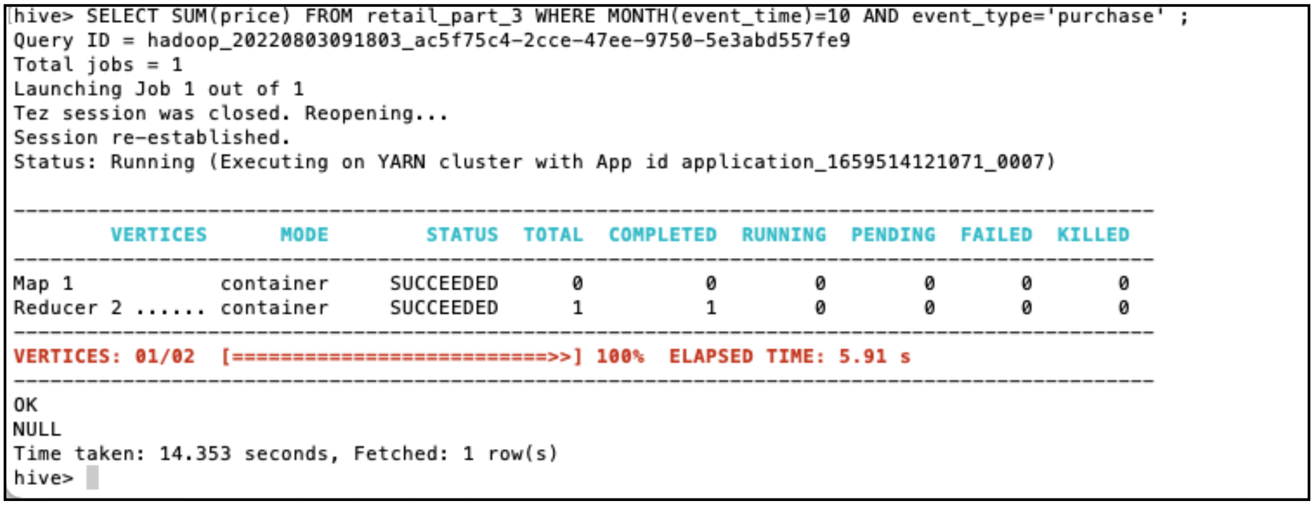
hive> CREATE EXTERNAL TABLE IF NOT EXISTS retail_part_1 (event_time timestamp, product_id string, category_id string, category_code string, brand string, price decimal(10,3), user_id bigint, user_session string) PARTITIONED BY(event_type string) CLUSTERED BY (user_id) INTO 5 buckets ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS textfile ;

hive> DESCRIBE retail_part_1 ;



```
hive> DESCRIBE retail_part_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: 0.106 seconds, Fetched: 14 row(s)
hive>
```

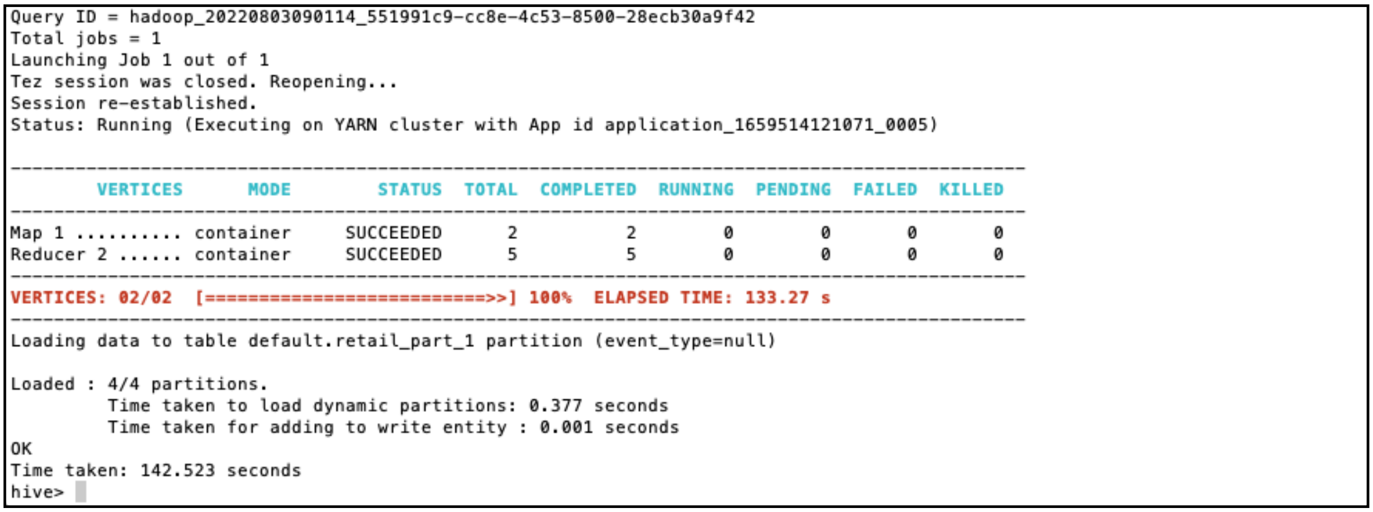
hive> INSERT INTO TABLE retail_part_1 PARTITION (event_type) SELECT event_time, product_id, category_id, category_code, brand, price, user_id, user_session, event_type FROM retail ;



```
hive> SELECT SUM(price) FROM retail_part_3 WHERE MONTH(event_time)=10 AND event_type='purchase' ;
Query ID = hadoop_20220803091803_ac5f75c4-2cce-47ee-9750-5e3abd557fe9
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_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1           container     SUCCEEDED      0          0        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 01/02  [==========================>>] 100%  ELAPSED TIME: 5.91 s
----------------------------------------------------------------------------------------------
OK
NULL
Time taken: 14.353 seconds, Fetched: 1 row(s)
hive>
```

Executing the same query with the new table “retail_part_1” to check the time.

hive> SELECT SUM(price) FROM retail_part_1 WHERE MONTH(event_time)=10 ANDevent_type='purchase' ;



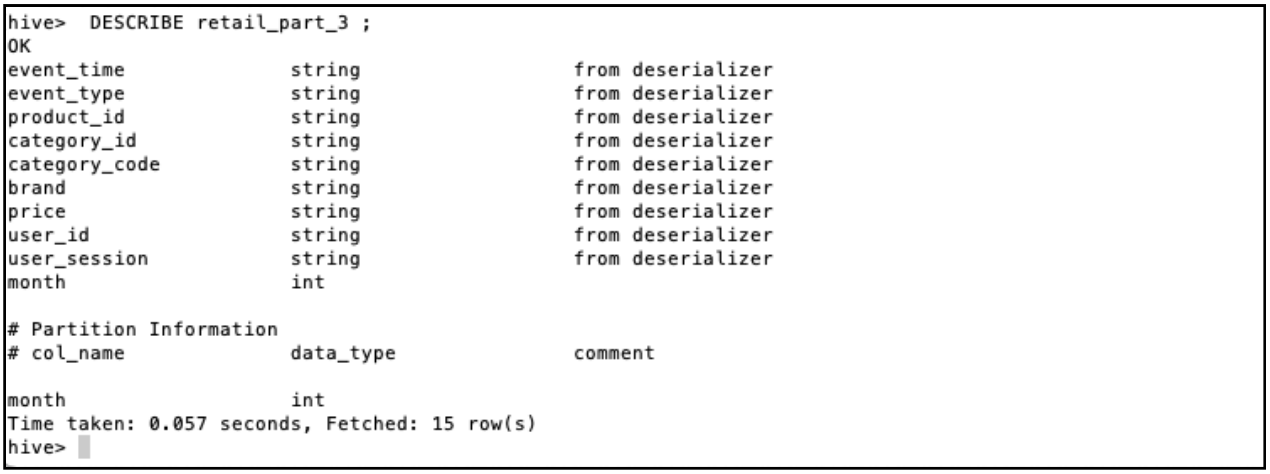
```
Query ID = hadoop_20220803090114_551991c9-cc8e-4c53-8500-28ecb30a9f42
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1659514121071_0005)

----------------------------------------------------------------------------------------------
        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: 133.27 s
----------------------------------------------------------------------------------------------
Loading data to table default.retail_part_1 partition (event_type=null)

Loaded : 4/4 partitions.
         Time taken to load dynamic partitions: 0.377 seconds
         Time taken for adding to write entity : 0.001 seconds
OK
Time taken: 142.523 seconds
hive>
```

PARTITION TABLE 2: retail_part_3

Partition on : month

hive> CREATE EXTERNAL TABLE IF NOT EXISTS retail_part_3 (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(month int) CLUSTERED BY (brand) INTO 5 buckets ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' STORED AS textfile ;

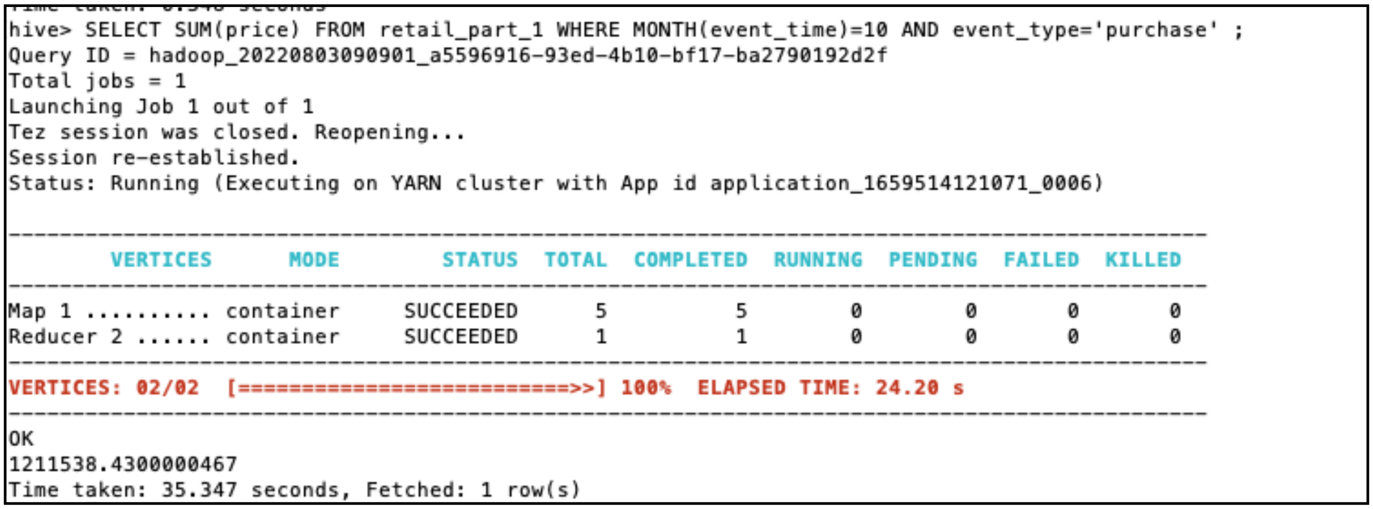
hive> DESCRIBE retail_part_3 ;



```
hive>  DESCRIBE retail_part_3 ;
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
month                   int

# Partition Information
# col_name              data_type               comment

month                   int
Time taken: 0.057 seconds, Fetched: 15 row(s)
hive>
```

Executing the same query with the new table "retail_part_3" to check the time.

hive> SELECT SUM(price) FROM retail_part_3 WHERE MONTH(event_time)=10 ANDevent_type='purchase' ;

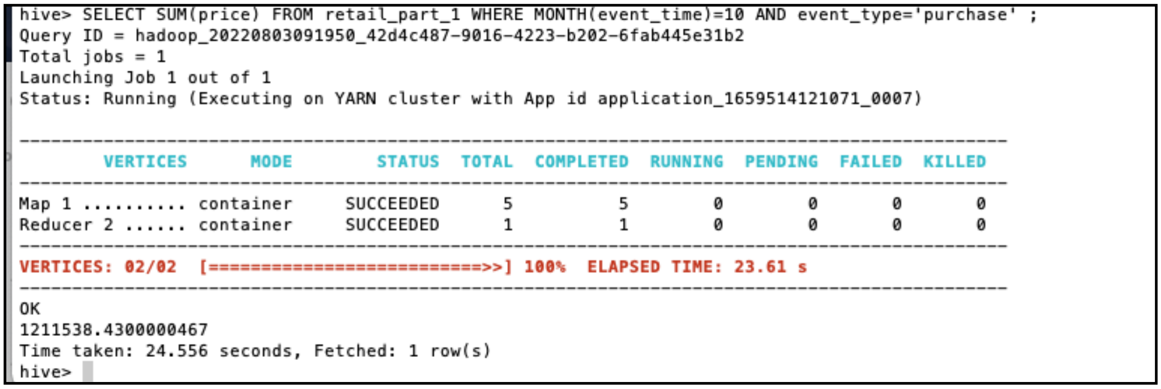


```
Time taken: 0.348 seconds
hive> SELECT SUM(price) FROM retail_part_1 WHERE MONTH(event_time)=10 AND event_type='purchase' ;
Query ID = hadoop_20220803090901_a5596916-93ed-4b10-bf17-ba2790192d2f
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1659514121071_0006)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 24.20 s
----------------------------------------------------------------------------------------------
OK
1211538.4300000467
Time taken: 35.347 seconds, Fetched: 1 row(s)
```

We get an optimised table by Partitioning on “event_type” and clustering by “user_id” . Hence, for all the following analysis, we will be using the optimised table “retail_part_1”.

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

hive> SELECT SUM(price) FROM retail_part_1 WHERE MONTH(event_time)=10 ANDevent_type='purchase' ;

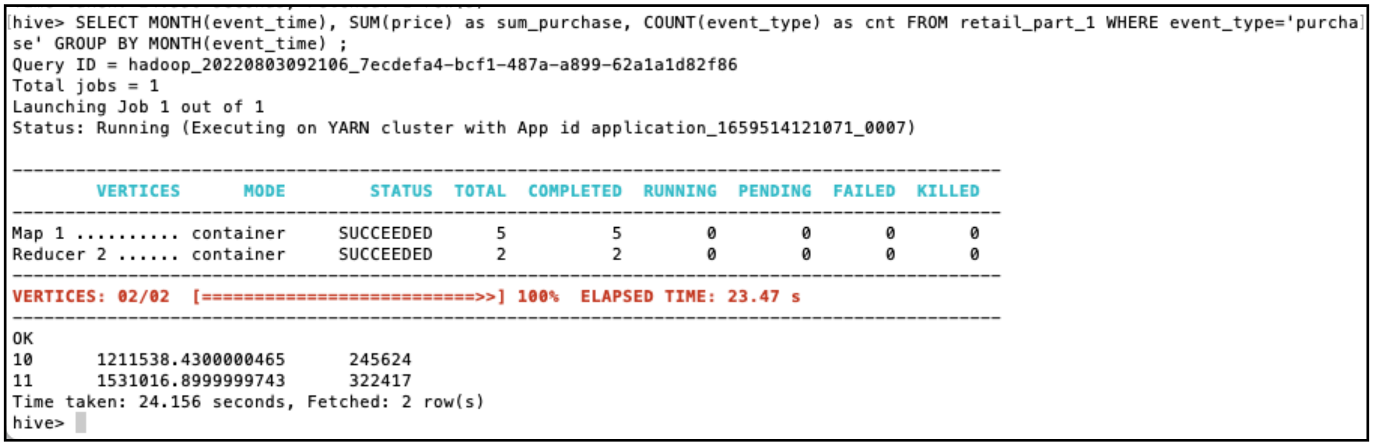


```
hive> SELECT SUM(price) FROM retail_part_1 WHERE MONTH(event_time)=10 AND event_type='purchase' ;
Query ID = hadoop_20220803091950_42d4c487-9016-4223-b202-6fab445e31b2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 23.61 s
----------------------------------------------------------------------------------------------
OK
1211538.4300000467
Time taken: 24.556 seconds, Fetched: 1 row(s)
hive>
```

Time Taken to execute the above query is 25.36 sec.

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

hive> SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROMretail_part_1 WHERE event_type='purchase' GROUP BY MONTH(event_time) ;

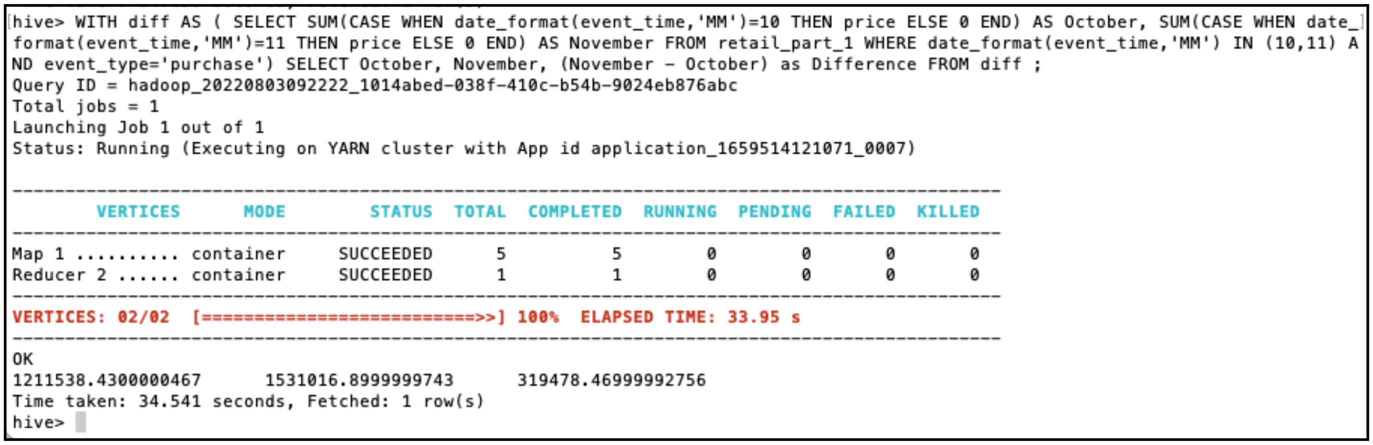


```
hive> SELECT MONTH(event_time), SUM(price) as sum_purchase, COUNT(event_type) as cnt FROM retail_part_1 WHERE event_type='purcha
se' GROUP BY MONTH(event_time) ;
Query ID = hadoop_20220803092106_7ecdefa4-bcf1-487a-a899-62a1a1d82f86
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      2          2        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 23.47 s
----------------------------------------------------------------------------------------------
OK
10      1211538.4300000465      245624
11      1531016.8999999743      322417
Time taken: 24.156 seconds, Fetched: 2 row(s)
hive>
```

In October month, 245624 purchases generated revenue of 1211538. Similarly in November month, 322417 purchases generated revenue of 1531016.8

Q) Write a query to find the change in revenue generated due to purchases from October toNovember.

hive>WITH diff AS (SELECT SUM(CASE WHEN date_format(event_time,'MM')=10 THEN price ELSE 0END) AS October, SUM(CASE WHEN date_format(event_time,'MM')=11 THEN price ELSE 0 END) AS November FROM retail_part_1 WHERE date_format(event_time,'MM') IN (10,11) AND event_type='purchase') SELECT October, November, (November - October) as Difference FROM diff ;

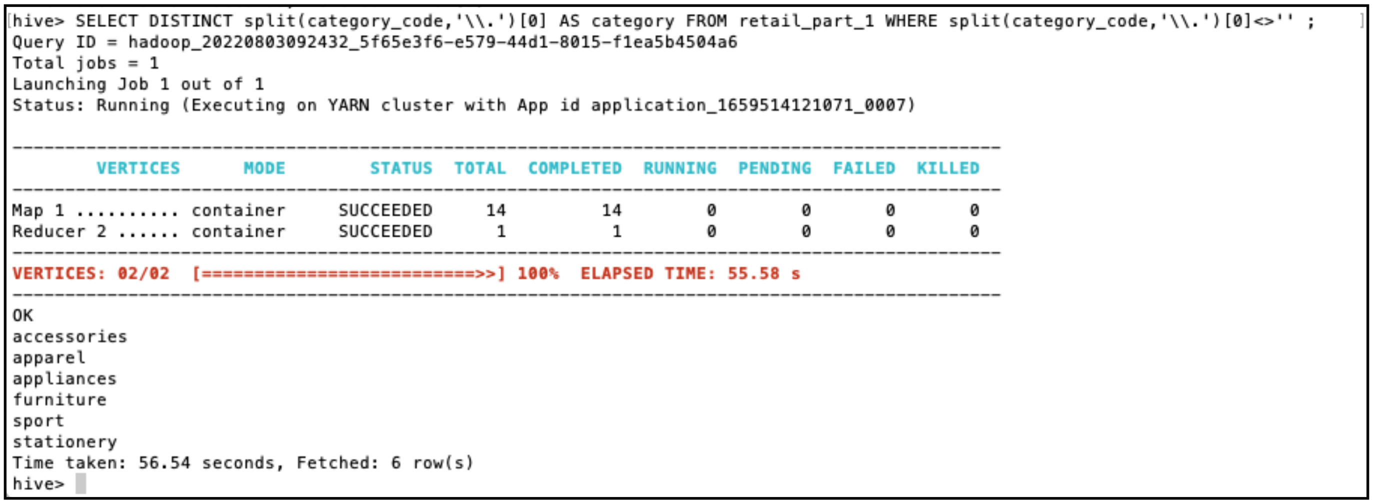


```
hive> WITH diff AS ( SELECT SUM(CASE WHEN date_format(event_time,'MM')=10 THEN price ELSE 0 END) AS October, SUM(CASE WHEN date_
format(event_time,'MM')=11 THEN price ELSE 0 END) AS November FROM retail_part_1 WHERE date_format(event_time,'MM') IN (10,11) A
ND event_type='purchase') SELECT October, November, (November - October) as Difference FROM diff ;
Query ID = hadoop_20220803092222_1014abed-038f-410c-b54b-9024eb876abc
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 33.95 s
----------------------------------------------------------------------------------------------
OK
1211538.4300000467      1531016.8999999743      319478.46999992756
Time taken: 34.541 seconds, Fetched: 1 row(s)
hive>
```

The change in revenue generated from October to November is 319478

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

hive>SELECT DISTINCT split(category_code,'\\.')[0] AS category FROM retail_part_1 WHEREsplit(category_code,'\\.')[0]<>" ;

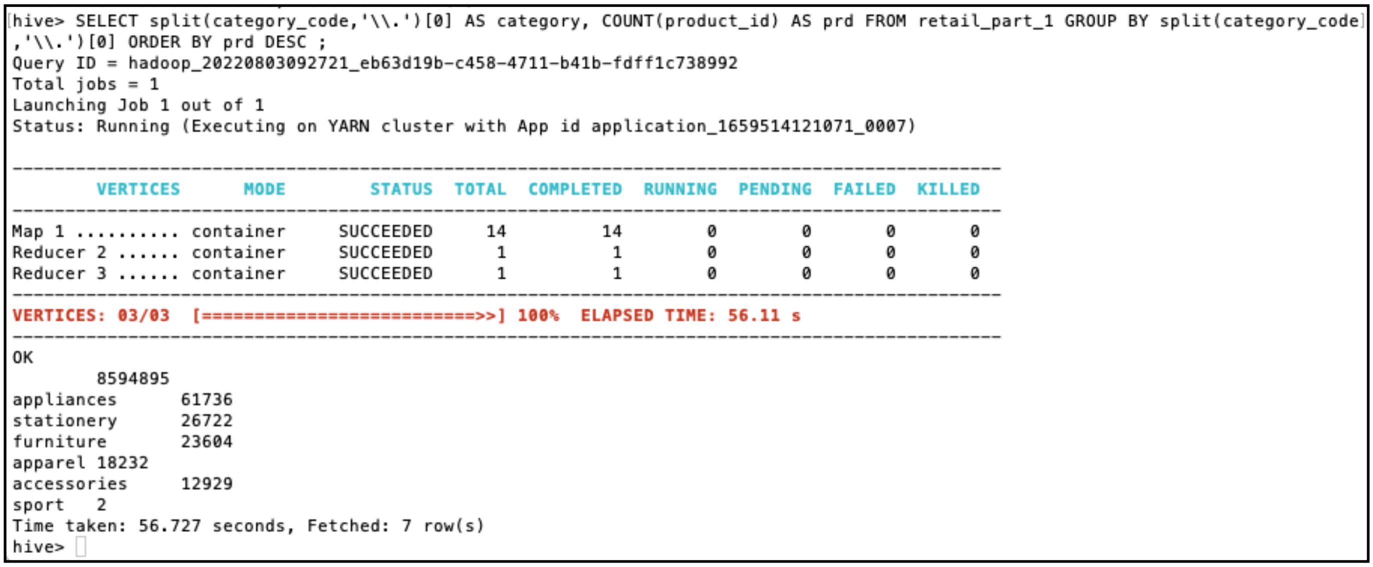


```
hive> SELECT DISTINCT split(category_code,'\\.')[0] AS category FROM retail_part_1 WHERE split(category_code,'\\.')[0]<>'' ;
Query ID = hadoop_20220803092432_5f65e3f6-e579-44d1-8015-f1ea5b4504a6
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED     14         14        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 55.58 s
----------------------------------------------------------------------------------------------
OK
accessories
apparel
appliances
furniture
sport
stationery
Time taken: 56.54 seconds, Fetched: 6 row(s)
hive>
```

There are 6 distinct categories of products. They are: Furniture, appliances, accessories, apparel, sport and stationary.

Q) Find the total number of products available under each category.

hive>SELECT split(category_code,'\\.')[0] AS category, COUNT(product_id) AS prd FROM retail_part_1 GROUP BY split(category_code,'\\.')[0] ORDER BY prd DESC ;

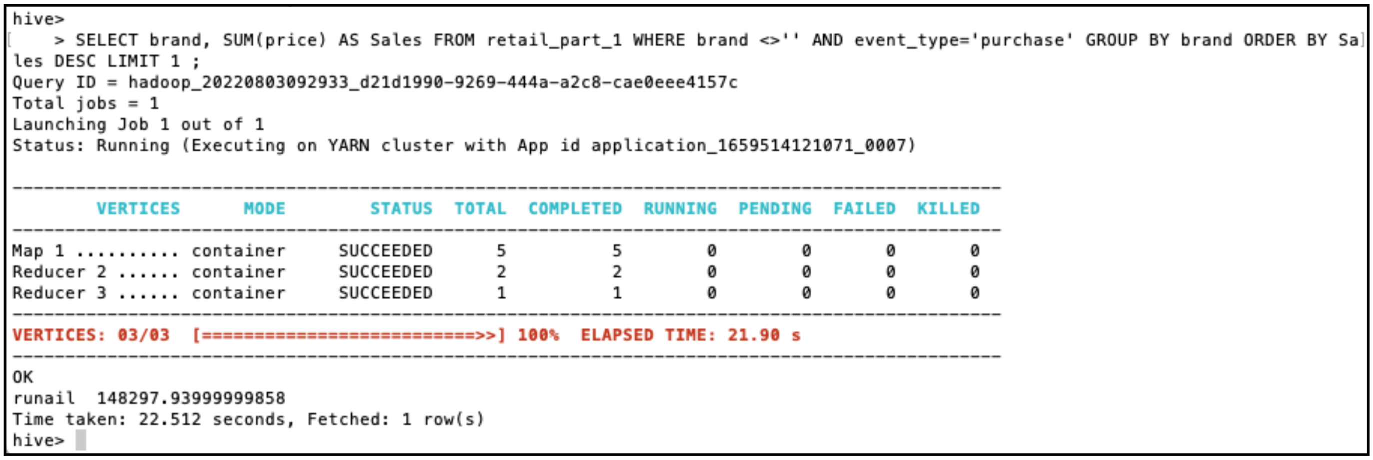


```
hive> SELECT split(category_code,'\\.')[0] AS category, COUNT(product_id) AS prd FROM retail_part_1 GROUP BY split(category_code
,'\\.')[0] ORDER BY prd DESC ;
Query ID = hadoop_20220803092721_eb63d19b-c458-4711-b41b-fdff1c738992
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED     14         14        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: 56.11 s
----------------------------------------------------------------------------------------------
OK
        8594895
appliances      61736
stationery      26722
furniture       23604
apparel 18232
accessories     12929
sport   2
Time taken: 56.727 seconds, Fetched: 7 row(s)
hive>
```

The sport category has the least number of products, whereas appliances has 61736 products.

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

SELECT brand, SUM(price) AS Sales FROM retail_part_1 WHERE brand <>'' AND event_type='purchase'GROUP BY brand ORDER BY Sales DESC LIMIT 1 ;

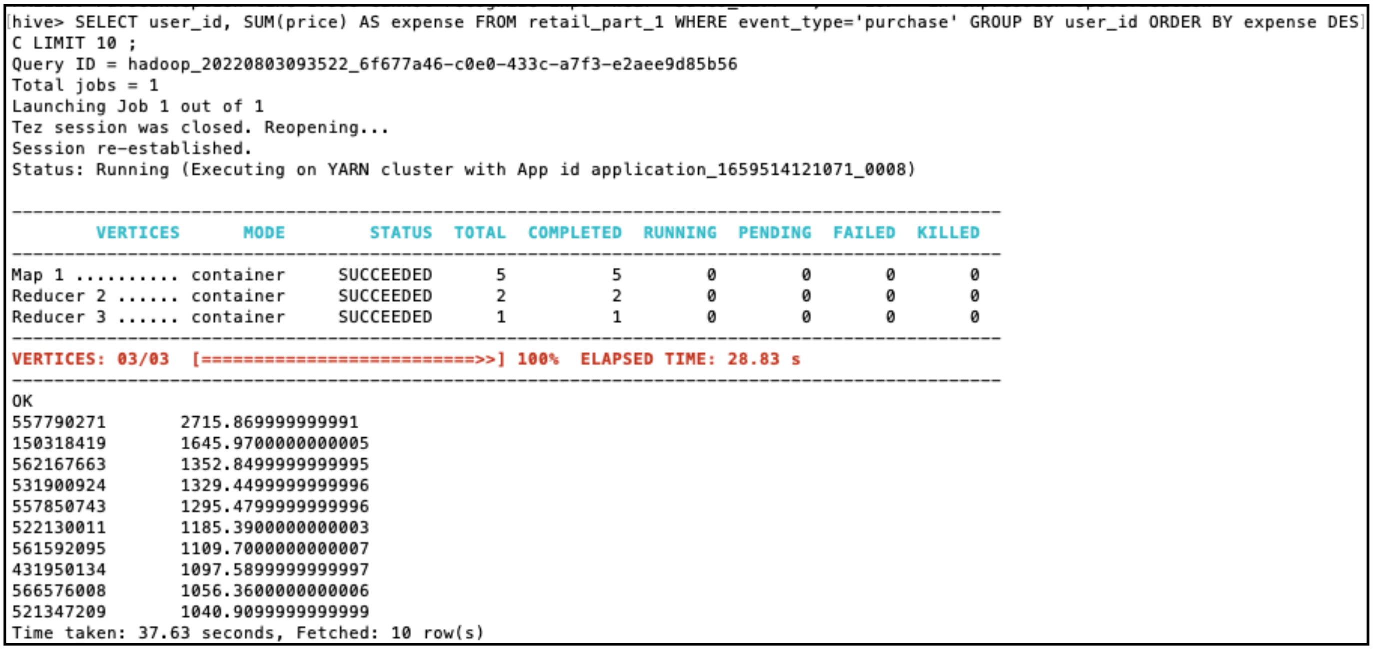


```
hive>
    > SELECT brand, SUM(price) AS Sales FROM retail_part_1 WHERE brand <>'' AND event_type='purchase' GROUP BY brand ORDER BY Sa
les DESC LIMIT 1 ;
Query ID = hadoop_20220803092933_d21d1990-9269-444a-a2c8-cae0eee4157c
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659514121071_0007)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        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: 21.90 s
----------------------------------------------------------------------------------------------
OK
runail  148297.93999999858
Time taken: 22.512 seconds, Fetched: 1 row(s)
hive>
```

Runail has the maximum sales for both months combined.

Q)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.

hive>SELECT user_id, SUM(price) AS expense FROM retail_part_1 WHERE event_type='purchase' GROUP BY user_id ORDER BY expense DESC LIMIT 10 ;

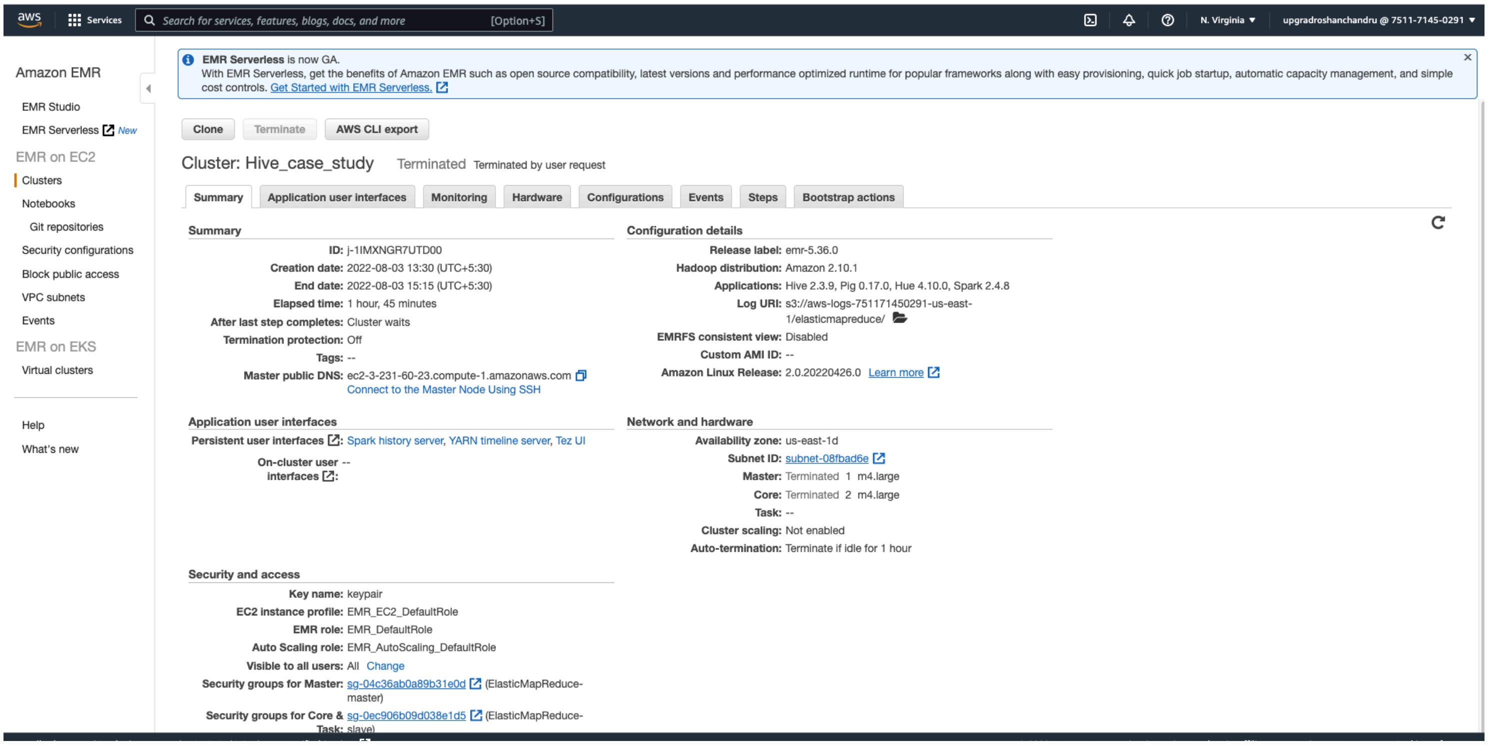


```
hive> SELECT user_id, SUM(price) AS expense FROM retail_part_1 WHERE event_type='purchase' GROUP BY user_id ORDER BY expense DES
C LIMIT 10 ;
Query ID = hadoop_20220803093522_6f677a46-c0e0-433c-a7f3-e2aee9d85b56
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1659514121071_0008)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        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: 28.83 s
----------------------------------------------------------------------------------------------
OK
557790271       2715.869999999991
150318419       1645.9700000000005
562167663       1352.8499999999995
531900924       1329.4499999999996
557850743       1295.4799999999996
522130011       1185.3900000000003
561592095       1109.7000000000007
431950134       1097.5899999999997
566576008       1056.3600000000006
521347209       1040.9099999999999
Time taken: 37.63 seconds, Fetched: 10 row(s)
```

Above we can find the list of the top 10 users who have spend the most.

Cleaning Up:

Once the analysis is completed, deleting the database & terminating the cluste



EMR Serverless is now GA.
With EMR Serverless, get the benefits of Amazon EMR such as open source compatibility, latest versions and performance optimized runtime for popular frameworks along with easy provisioning, quick job startup, automatic capacity management, and simple cost controls. Get Started with EMR Serverless.

Cluster: Hive_case_study Terminated Terminated by user request

Summary

ID: j-1IMXNGR7UTD00
Creation date: 2022-08-03 13:30 (UTC+5:30)
End date: 2022-08-03 15:15 (UTC+5:30)
Elapsed time: 1 hour, 45 minutes
After last step completes: Cluster waits
Termination protection: Off
Tags: --
Master public DNS: ec2-3-231-60-23.compute-1.amazonaws.com
Connect to the Master Node Using SSH

Configuration details

Release label: emr-5.36.0
Hadoop distribution: Amazon 2.10.1
Applications: Hive 2.3.9, Pig 0.17.0, Hue 4.10.0, Spark 2.4.8
Log URI: s3://aws-logs-751171450291-us-east-1/elasticmapreduce/
EMRFS consistent view: Disabled
Custom AMI ID: --
Amazon Linux Release: 2.0.20220426.0 Learn more

Application user interfaces

Persistent user interfaces: Spark history server, YARN timeline server, Tez UI
On-cluster user interfaces: --

Network and hardware

Availability zone: us-east-1d
Subnet ID: subnet-08fbad6e
Master: Terminated 1 m4.large
Core: Terminated 2 m4.large
Task: --
Cluster scaling: Not enabled
Auto-termination: Terminate if idle for 1 hour

Security and access

Key name: keypair
EC2 instance profile: EMR_EC2_DefaultRole
EMR role: EMR_DefaultRole
Auto Scaling role: EMR_AutoScaling_DefaultRole
Visible to all users: All Change
Security groups for Master: sg-04c36ab0a89b31e0d (ElasticMapReduce-master)
Security groups for Core & Task: sg-0ec906b09d038e1d5 (ElasticMapReduce-slave)