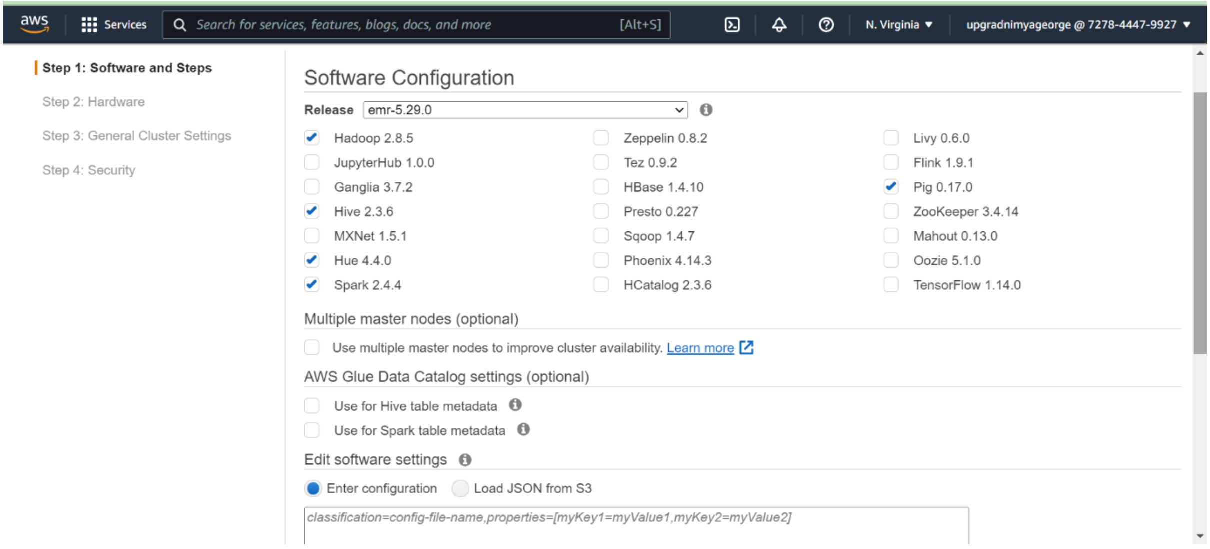
HIVE CASE STUDY

EMR CLUSTER CREATION

- Login to your AWS account & search EMR services. After the EMR home page appears click on Create cluster & follow the steps as mentioned. We have chosen cluster release version 5.29.0 in our case study.



aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
upgradnimyageorge @ 7278-4447-9927

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

Software Configuration

Release: emr-5.29.0

☑ Hadoop 2.8.5	□ Zeppelin 0.8.2	□ Livy 0.6.0
□ JupyterHub 1.0.0	□ Tez 0.9.2	□ Flink 1.9.1
□ Ganglia 3.7.2	□ HBase 1.4.10	☑ Pig 0.17.0
☑ Hive 2.3.6	□ Presto 0.227	□ ZooKeeper 3.4.14
□ MXNet 1.5.1	□ Sqoop 1.4.7	□ Mahout 0.13.0
☑ Hue 4.4.0	□ Phoenix 4.14.3	□ Oozie 5.1.0
☑ Spark 2.4.4	□ HCatalog 2.3.6	□ TensorFlow 1.14.0

Multiple master nodes (optional)

□ Use multiple master nodes to improve cluster availability. Learn more

AWS Glue Data Catalog settings (optional)

□ Use for Hive table metadata

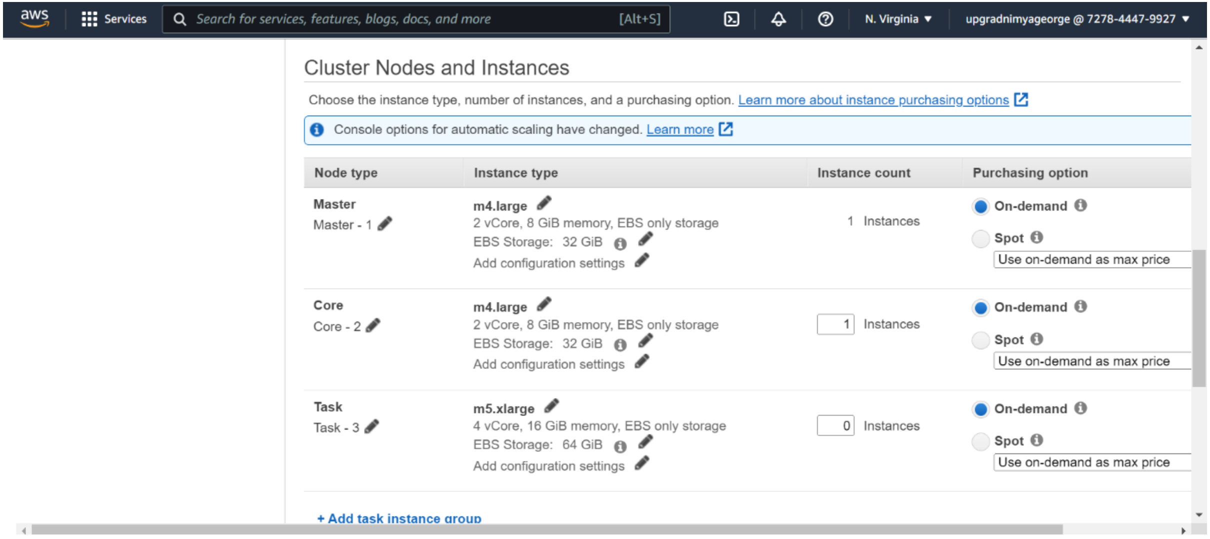
□ Use for Spark table metadata

Edit software settings

● Enter configuration ○ Load JSON from S3

classification=config-file-name,properties=[myKey1=myValue1,myKey2=myValue2]

- We will be going for a 2-node cluster for our analysis & we will select m4.large instance type each for both master & core node.



aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
upgradnimyageorge @ 7278-4447-9927

Cluster Nodes and Instances

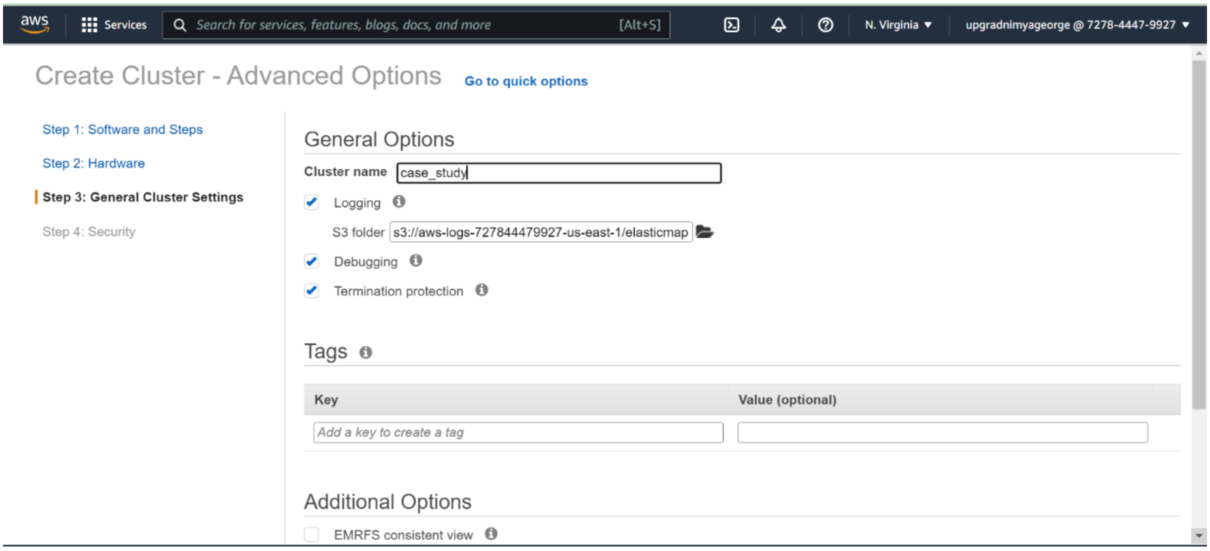
Choose the instance type, number of instances, and a purchasing option. Learn more about instance purchasing options

Console options for automatic scaling have changed. Learn more

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	1 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: 64 GiB Add configuration settings	0 Instances	● On-demand ○ Spot Use on-demand as max price

+ Add task instance group

- Select a cluster name. Here we have taken the cluster name as case_study.


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
upgradnimyageorge @ 7278-4447-9927 ▼
Go to quick options
Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security

Create Cluster - Advanced Options

General Options

Cluster name: case_study

☑ Logging

S3 folder: s3://aws-logs-727844479927-us-east-1/elasticmap

☑ Debugging

☑ Termination protection

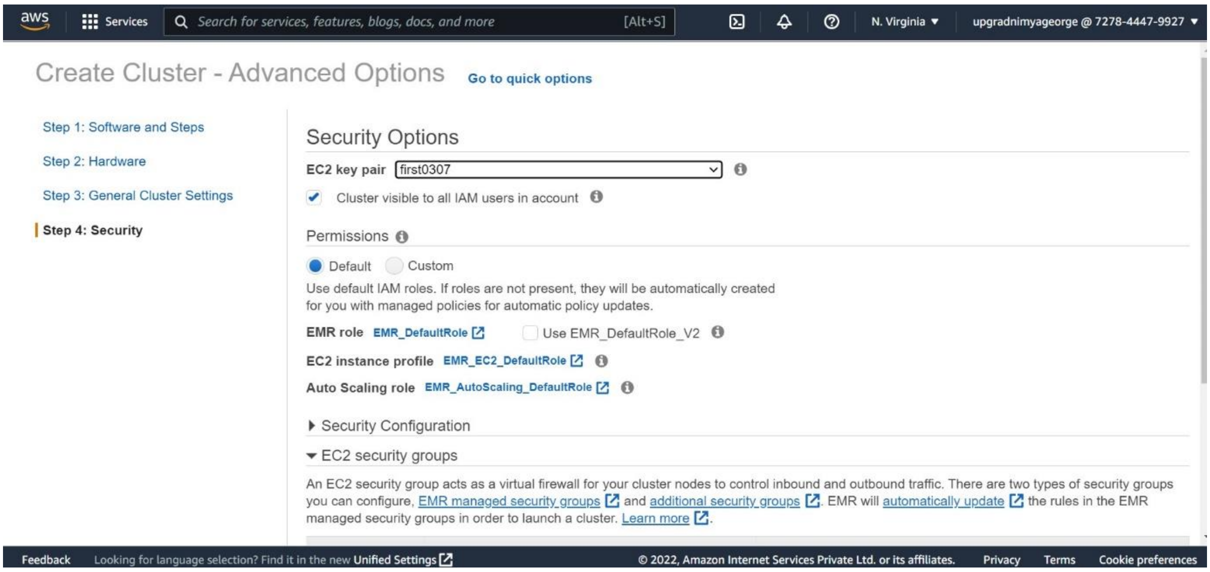
Tags

Key	Value (optional)
Add a key to create a tag	

Additional Options

□ EMRFS consistent view

- Select an already created key-pair which will be used while connecting to master node.


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
upgradnimyageorge @ 7278-4447-9927 ▼
Go to quick options
Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

Create Cluster - Advanced Options

Security Options

EC2 key pair: first0307

☑ Cluster visible to all IAM users in account

Permissions

● Default ○ Custom

Use default IAM roles. If roles are not present, they will be automatically created for you with managed policies for automatic policy updates.

EMR role: EMR_DefaultRole □ Use EMR_DefaultRole_V2

EC2 instance profile: EMR_EC2_DefaultRole

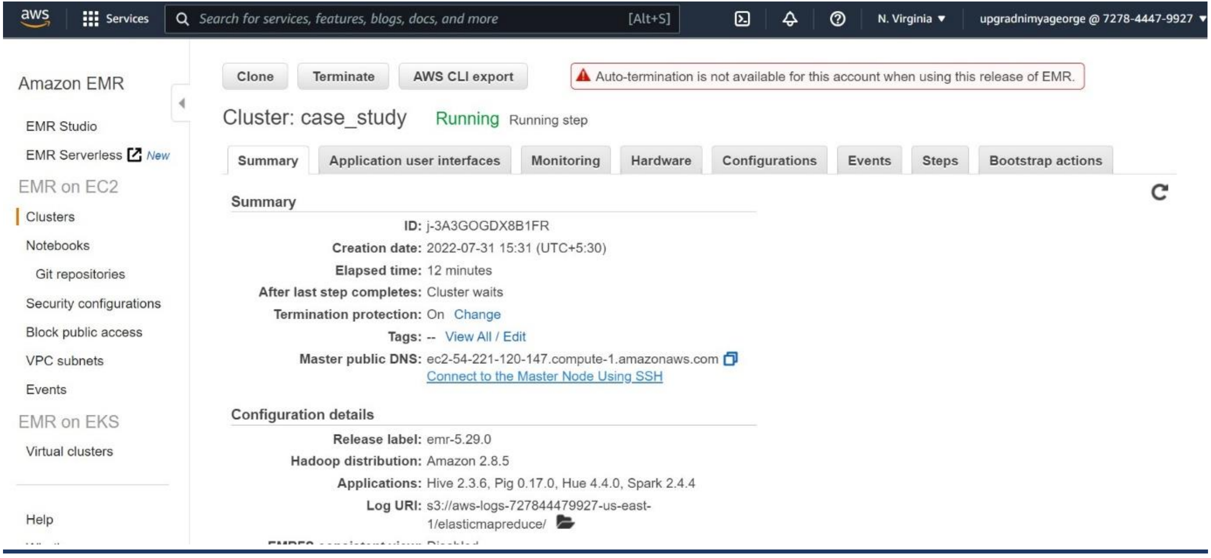
Auto Scaling role: EMR_AutoScaling_DefaultRole

▸ Security Configuration

▾ EC2 security groups

An EC2 security group acts as a virtual firewall for your cluster nodes to control inbound and outbound traffic. There are two types of security groups you can configure, EMR managed security groups and additional security groups. EMR will automatically update the rules in the EMR managed security groups in order to launch a cluster. Learn more.

- Our cluster has been created successfully and is in running state which indicates its ready to be connected from the local system.



Auto-termination is not available for this account when using this release of EMR.

Cluster: case_study Running Running step

Summary

ID: j-3A3GOGDX8B1FR

Creation date: 2022-07-31 15:31 (UTC+5:30)

Elapsed time: 12 minutes

After last step completes: Cluster waits

Termination protection: On Change

Tags: -- View All / Edit

Master public DNS: ec2-54-221-120-147.compute-1.amazonaws.com

Connect to the Master Node Using SSH

Configuration details

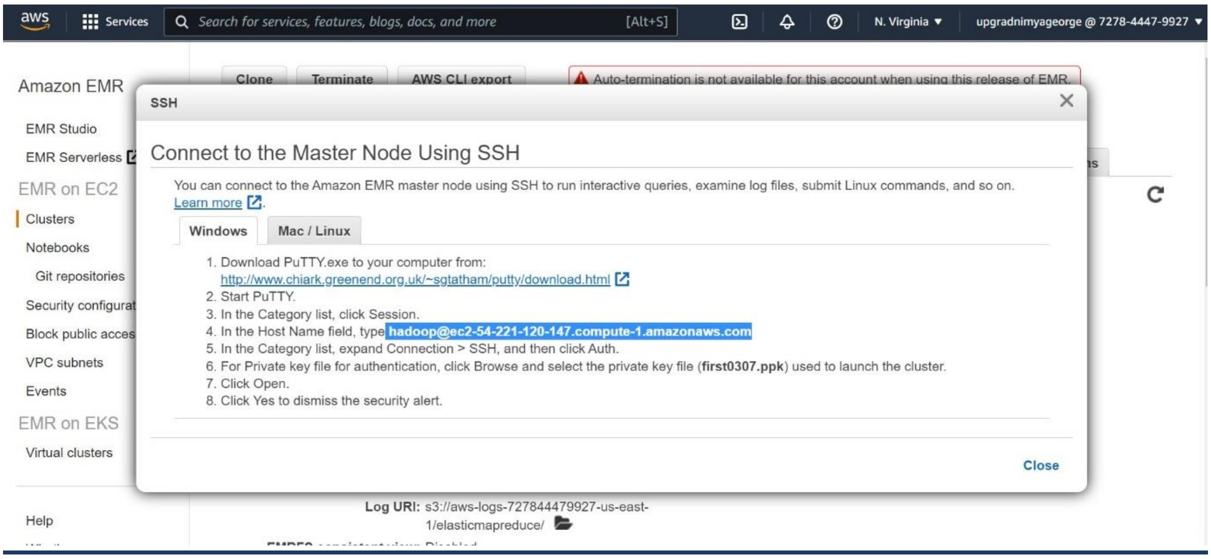
Release label: emr-5.29.0

Hadoop distribution: Amazon 2.8.5

Applications: Hive 2.3.6, Pig 0.17.0, Hue 4.4.0, Spark 2.4.4

Log URI: s3://aws-logs-727844479927-us-east-1/elasticmapreduce/

- Copy the highlighted link i.e., the master public DNS.Next, paste the address in the Host Name field.



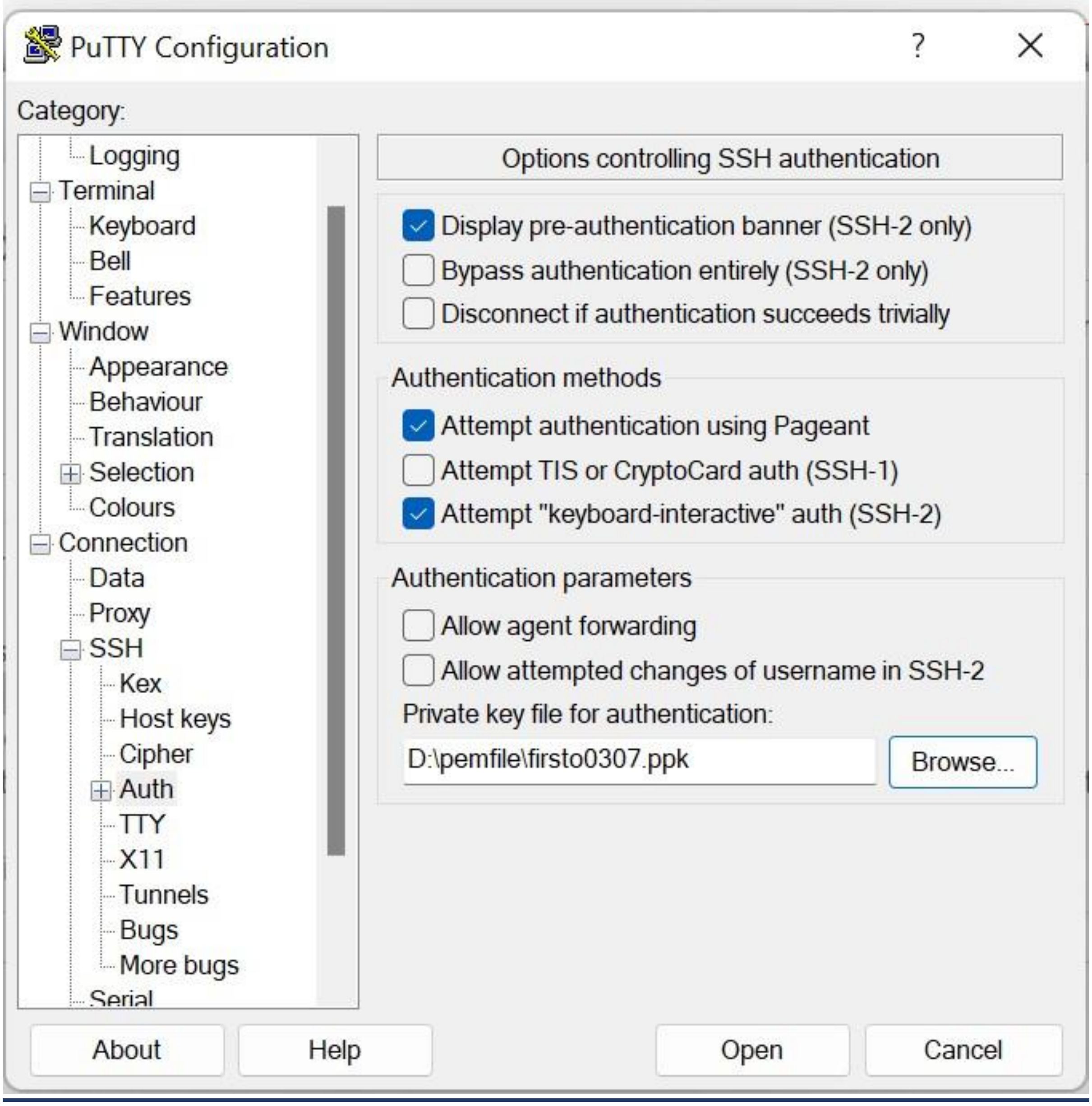
Connect to the Master Node Using SSH

You can connect to the Amazon EMR master node using SSH to run interactive queries, examine log files, submit Linux commands, and so on. Learn more.

1. Download PuTTY.exe to your computer from: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
2. Start PuTTY.
3. In the Category list, click Session.
4. In the Host Name field, type hadoop@ec2-54-221-120-147.compute-1.amazonaws.com
5. In the Category list, expand Connection > SSH, and then click Auth.
6. For Private key file for authentication, click Browse and select the private key file (**first0307.ppk**) used to launch the cluster.
7. Click Open.
8. Click Yes to dismiss the security alert.

Log URI: s3://aws-logs-727844479927-us-east-1/elasticmapreduce/

- Click on SSH & then Auth. Give the location where the key pair is stored in the local system and click on Open.



Options controlling SSH authentication

☑ Display pre-authentication banner (SSH-2 only)
□ Bypass authentication entirely (SSH-2 only)
□ Disconnect if authentication succeeds trivially

Authentication methods

☑ Attempt authentication using Pageant
□ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

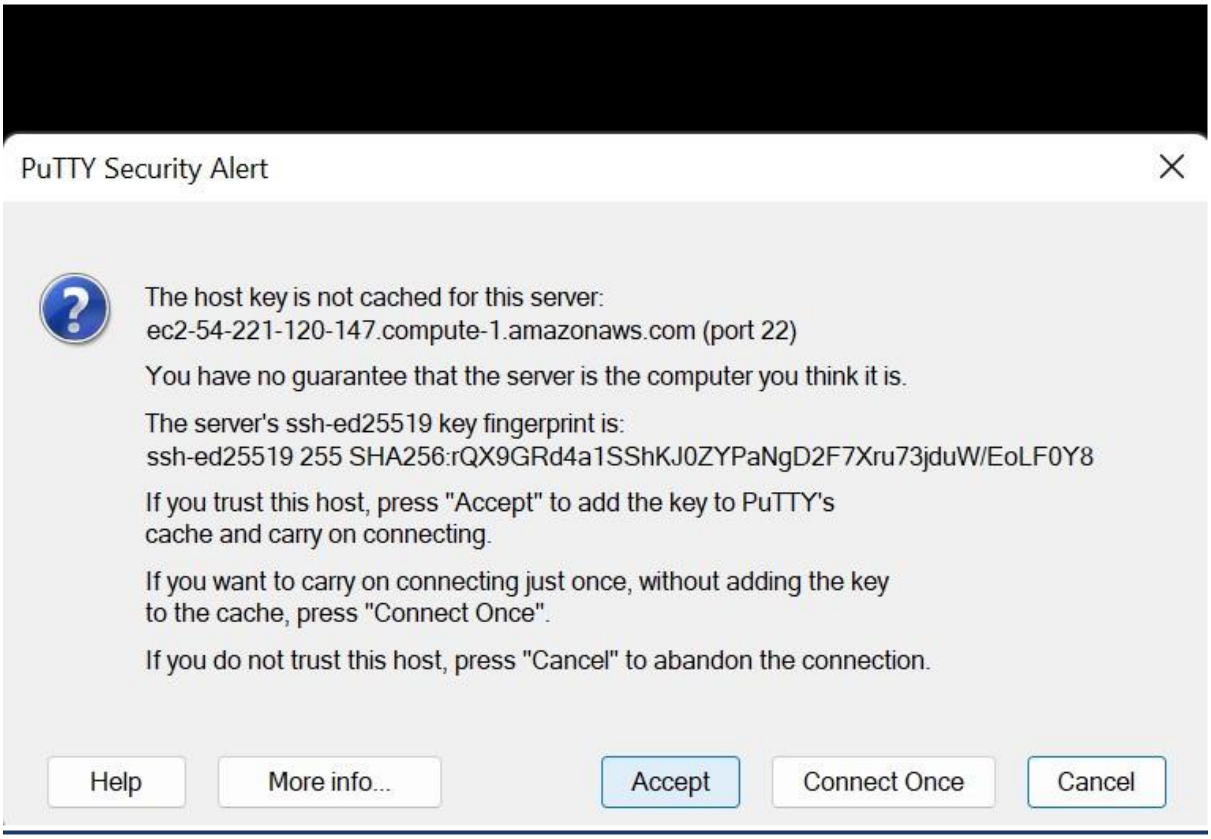
Authentication parameters

□ Allow agent forwarding
□ Allow attempted changes of username in SSH-2

Private key file for authentication:

D:\pemfile\firsto0307.ppk Browse...

- Next click on Accept which will open the SSH terminal. After we have created an EMR cluster & successfully connected to it via putty we can begin to code in the Terminal.



PuTTY Security Alert

The host key is not cached for this server:
ec2-54-221-120-147.compute-1.amazonaws.com (port 22)

You have no guarantee that the server is the computer you think it is.

The server's ssh-ed25519 key fingerprint is:
ssh-ed25519 255 SHA256:rQX9GRd4a1SShKJ0ZYPaNgD2F7Xru73jduW/EoLF0Y8

If you trust this host, press "Accept" to add the key to PuTTY's cache and carry on connecting.

If you want to carry on connecting just once, without adding the key to the cache, press "Connect Once".

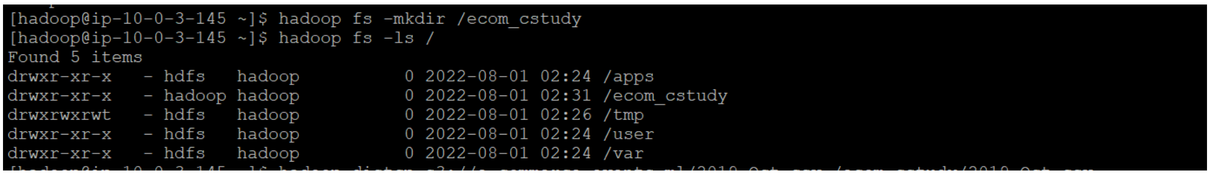
If you do not trust this host, press "Cancel" to abandon the connection.

WORKING WITH HDFS :

Creating a folder in Hadoop

hadoop fs -mkdir /ecom_cstudy

hadoop fs -ls /

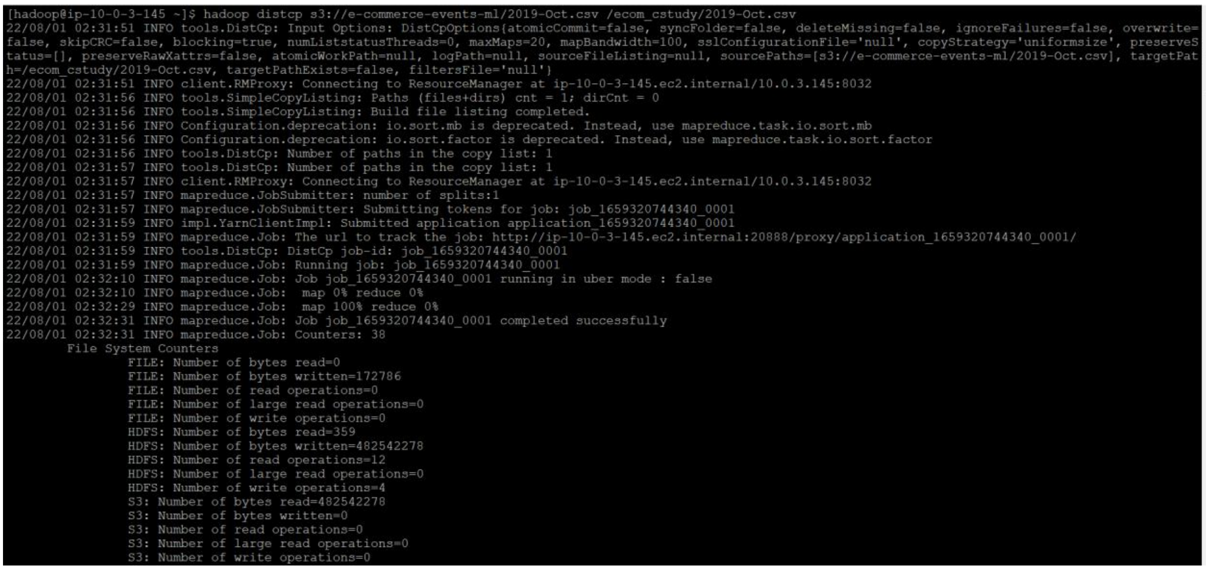


```
[hadoop@ip-10-0-3-145 ~]$ hadoop fs -mkdir /ecom_cstudy
[hadoop@ip-10-0-3-145 ~]$ hadoop fs -ls /
Found 5 items
drwxr-xr-x   - hdfs   hadoop          0 2022-08-01 02:24 /apps
drwxr-xr-x   - hadoop hadoop          0 2022-08-01 02:31 /ecom_cstudy
drwxrwxrwt   - hdfs   hadoop          0 2022-08-01 02:26 /tmp
drwxr-xr-x   - hdfs   hadoop          0 2022-08-01 02:24 /user
drwxr-xr-x   - hdfs   hadoop          0 2022-08-01 02:24 /var
```

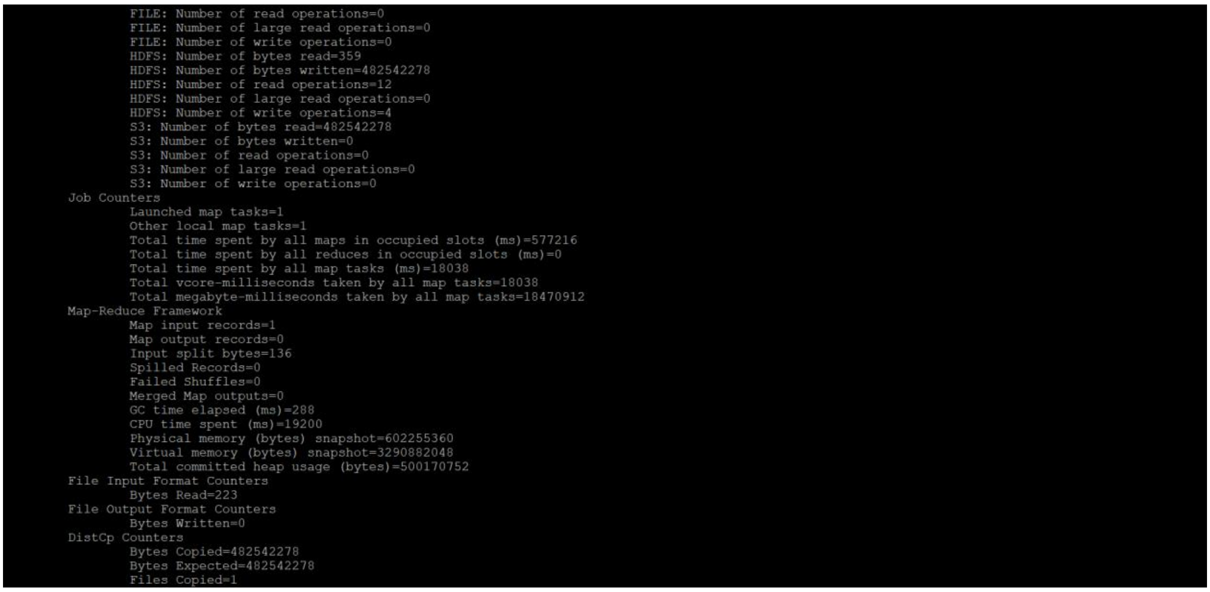
Copying October & November data from S3 bucket into HDFS .

hadoop distcp s3://e-commerce-events-ml/2019-Oct.csv /ecom_cstudy/2019-Oct.csv

hadoop distcp s3://e-commerce-events-ml/2019-Nov.csv /ecom_cstudy/2019-Nov.csv



```
[hadoop@ip-10-0-3-145 ~]$ hadoop distcp s3://e-commerce-events-ml/2019-Oct.csv /ecom_cstudy/2019-Oct.csv
22/08/01 02:31:51 INFO tools.DistCp: Input Options: DistCpOptions{atomicCommit=false, syncFolder=false, deleteMissing=false, ignoreFailures=false, overwrite=
false, skipCRC=false, blocking=true, numListstatusThreads=0, maxMaps=20, mapBandwidth=100, sslConfigurationFile='null', copyStrategy='uniformsize', preserveS
tatus=[], preserveRawXattrs=false, atomicWorkPath=null, logPath=null, sourceFileListing=null, sourcePaths=[s3://e-commerce-events-ml/2019-Oct.csv], targetPat
h=/ecom_cstudy/2019-Oct.csv, targetPathExists=false, filtersFile='null'}
22/08/01 02:31:51 INFO client.RMProxy: Connecting to ResourceManager at ip-10-0-3-145.ec2.internal/10.0.3.145:8032
22/08/01 02:31:56 INFO tools.SimpleCopyListing: Paths (files+dirs) cnt = 1; dirCnt = 0
22/08/01 02:31:56 INFO tools.SimpleCopyListing: Build file listing completed.
22/08/01 02:31:56 INFO Configuration.deprecation: io.sort.mb is deprecated. Instead, use mapreduce.task.io.sort.mb
22/08/01 02:31:56 INFO Configuration.deprecation: io.sort.factor is deprecated. Instead, use mapreduce.task.io.sort.factor
22/08/01 02:31:56 INFO tools.DistCp: Number of paths in the copy list: 1
22/08/01 02:31:57 INFO tools.DistCp: Number of paths in the copy list: 1
22/08/01 02:31:57 INFO client.RMProxy: Connecting to ResourceManager at ip-10-0-3-145.ec2.internal/10.0.3.145:8032
22/08/01 02:31:57 INFO mapreduce.JobSubmitter: number of splits:1
22/08/01 02:31:57 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1659320744340_0001
22/08/01 02:31:59 INFO impl.YarnClientImpl: Submitted application application_1659320744340_0001
22/08/01 02:31:59 INFO mapreduce.Job: The url to track the job: http://ip-10-0-3-145.ec2.internal:20888/proxy/application_1659320744340_0001/
22/08/01 02:31:59 INFO tools.DistCp: DistCp job-id: job_1659320744340_0001
22/08/01 02:31:59 INFO mapreduce.Job: Running job: job_1659320744340_0001
22/08/01 02:32:10 INFO mapreduce.Job: Job job_1659320744340_0001 running in uber mode : false
22/08/01 02:32:10 INFO mapreduce.Job:  map 0% reduce 0%
22/08/01 02:32:29 INFO mapreduce.Job:  map 100% reduce 0%
22/08/01 02:32:31 INFO mapreduce.Job: Job job_1659320744340_0001 completed successfully
22/08/01 02:32:31 INFO mapreduce.Job: Counters: 38
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=172786
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=359
                HDFS: Number of bytes written=482542278
                HDFS: Number of read operations=12
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=4
                S3: Number of bytes read=482542278
                S3: Number of bytes written=0
                S3: Number of read operations=0
                S3: Number of large read operations=0
                S3: Number of write operations=0
```

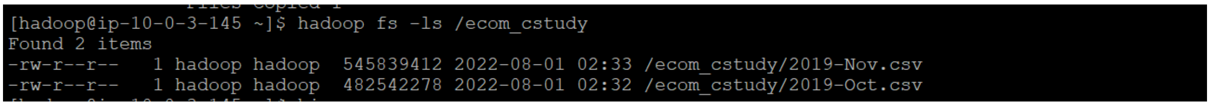


```
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=359
                HDFS: Number of bytes written=482542278
                HDFS: Number of read operations=12
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=4
                S3: Number of bytes read=482542278
                S3: Number of bytes written=0
                S3: Number of read operations=0
                S3: Number of large read operations=0
                S3: Number of write operations=0
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=577216
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=18038
                Total vcore-milliseconds taken by all map tasks=18038
                Total megabyte-milliseconds taken by all map tasks=18470912
        Map-Reduce Framework
                Map input records=1
                Map output records=0
                Input split bytes=136
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=288
                CPU time spent (ms)=19200
                Physical memory (bytes) snapshot=602255360
                Virtual memory (bytes) snapshot=3290882048
                Total committed heap usage (bytes)=500170752
        File Input Format Counters
                Bytes Read=223
        File Output Format Counters
                Bytes Written=0
        DistCp Counters
                Bytes Copied=482542278
                Bytes Expected=482542278
                Files Copied=1
```

The same way we will upload data of November month.

Verifying if data has been copied successfully.

hadoop fs -ls /ecom_cstudy

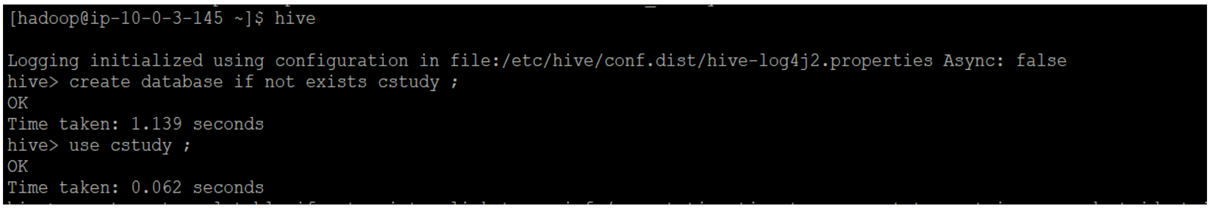


```
[hadoop@ip-10-0-3-145 ~]$ hadoop fs -ls /ecom_cstudy
Found 2 items
-rw-r--r--   1 hadoop hadoop  545839412 2022-08-01 02:33 /ecom_cstudy/2019-Nov.csv
-rw-r--r--   1 hadoop hadoop  482542278 2022-08-01 02:32 /ecom_cstudy/2019-Oct.csv
```

Working on hive.

create database if not exists cstudy ;

use cstudy ;

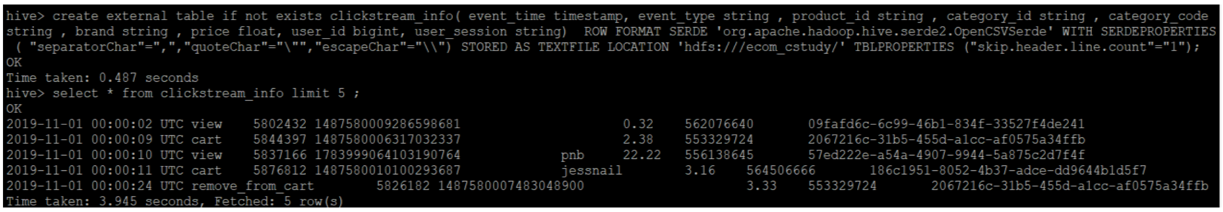


```
[hadoop@ip-10-0-3-145 ~]$ hive

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> create database if not exists cstudy ;
OK
Time taken: 1.139 seconds
hive> use cstudy ;
OK
Time taken: 0.062 seconds
```

Creating a common table named clickstream and storing both October & November data in it.

create external table if not exists clickstream_info(event_time timestamp, event_type string , product_id string , category_id string , category_code string , brand string , price float, user_id bigint, user_session string) ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH SERDEPROPERTIES ("separatorChar"=",","quoteChar"="\"","escapeChar"="\\") STORED AS TEXTFILE LOCATION 'hdfs:///ecom_cstudy/' TBLPROPERTIES ("skip.header.line.count"="1");

select * from clickstream_info limit 5 ;



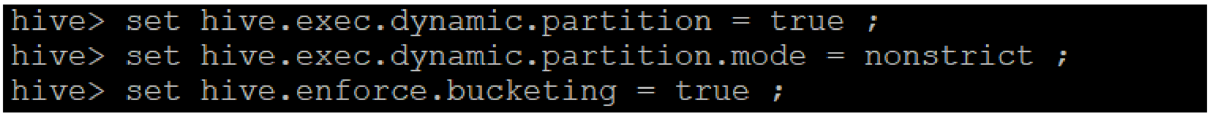
```
hive> create external table if not exists clickstream_info( event_time timestamp, event_type string , product_id string , category_id string , category_code
string , brand string , price float, user_id bigint, user_session string)  ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde' WITH SERDEPROPERTIES
 ( "separatorChar"=",","quoteChar"="\"","escapeChar"="\\") STORED AS TEXTFILE LOCATION 'hdfs:///ecom_cstudy/' TBLPROPERTIES ("skip.header.line.count"="1");
OK
Time taken: 0.487 seconds
hive> select * from clickstream_info 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     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
Time taken: 3.945 seconds, Fetched: 5 row(s)
```

To create optimised table having partitions & buckets we need to enable some settings:

set hive.exec.dynamic.partition = true ;

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

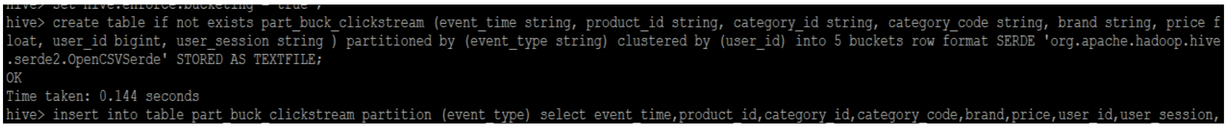
set hive.enforce.bucketing = true ;



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

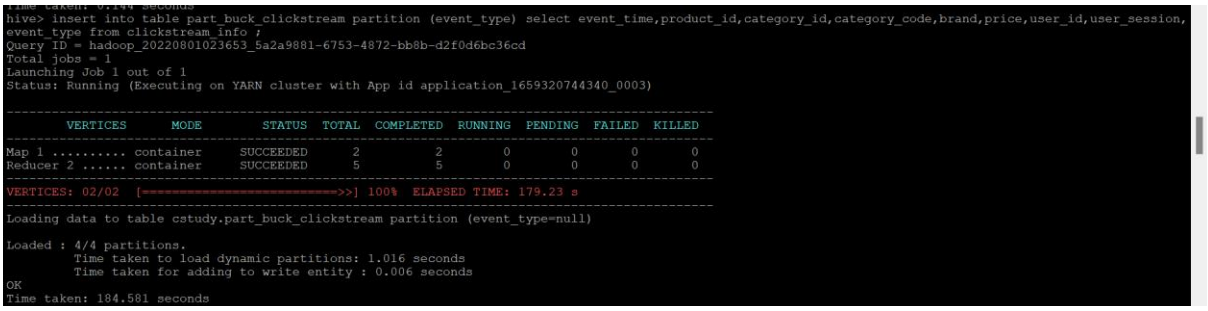
Creating table with partitions and buckets and inserting data into it.

create table if not exists part_buck_clickstream (event_time string, product_id string, category_id string, category_code string, brand string, price float, user_id bigint, user_session string) partitioned by (event_type string) clustered by (category_code) into 13 buckets row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile;



```
hive> set hive.enforce.bucketing = true ;
hive> create table if not exists part_buck_clickstream (event_time string, product_id string, category_id string, category_code string, brand string, price f
loat, 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;
OK
Time taken: 0.144 seconds
hive> insert into table part_buck_clickstream partition (event_type) select event_time,product_id,category_id,category_code,brand,price,user_id,user_session,
```

insert into table part_buck_clickstream partition (event_type) select event_time,product_id,category_id,category_code,brand,price,user_id,user_session,event_type from clickstream_info ;

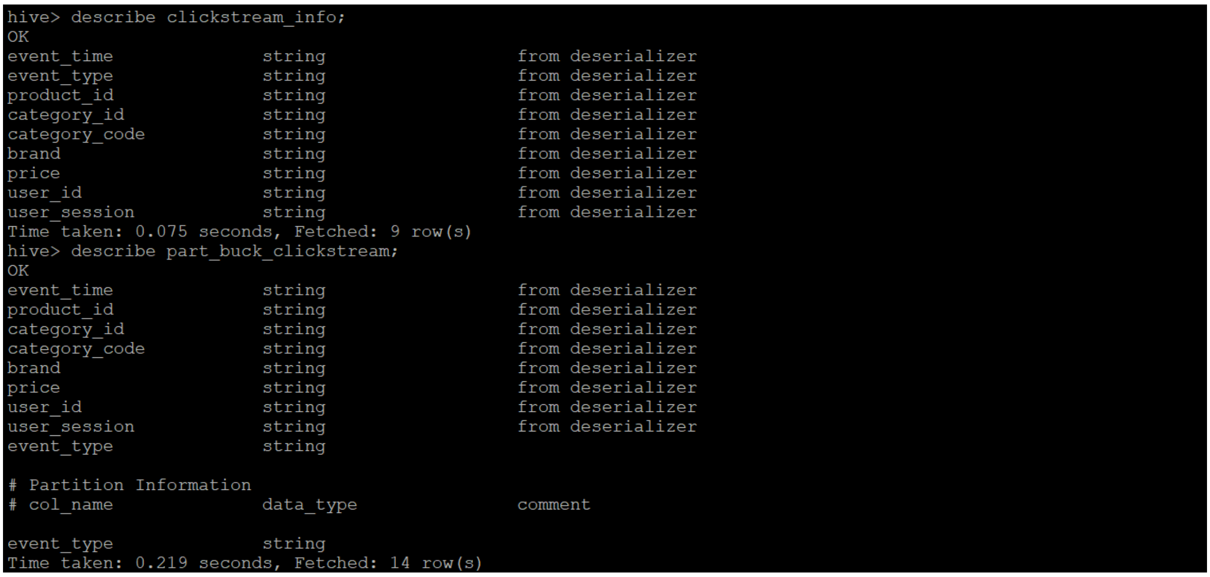


```
Time taken: 0.144 seconds
hive> insert into table part_buck_clickstream partition (event_type) select event_time,product_id,category_id,category_code,brand,price,user_id,user_session,
event_type from clickstream_info ;
Query ID = hadoop_20220801023653_5a2a9881-6753-4872-bb8b-d2f0d6bc36cd
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 179.23 s
----------------------------------------------------------------------------------------------
Loading data to table cstudy.part_buck_clickstream partition (event_type=null)

Loaded : 4/4 partitions.
         Time taken to load dynamic partitions: 1.016 seconds
         Time taken for adding to write entity : 0.006 seconds
OK
Time taken: 184.581 seconds
```

Describing both tables.

describe clickstream_info;

describe part_buck_clickstream;



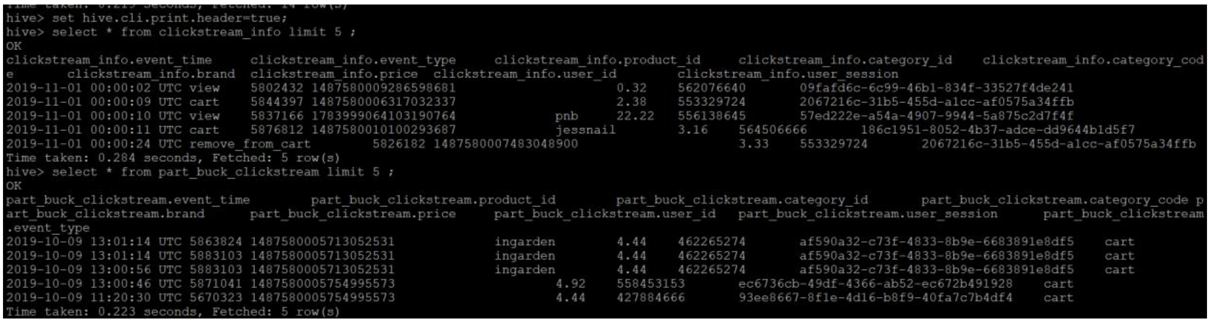
```
hive> describe clickstream_info;
OK
event_time              string                  from deserializer
event_type              string                  from deserializer
product_id              string                  from deserializer
category_id             string                  from deserializer
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: 0.075 seconds, Fetched: 9 row(s)
hive> describe part_buck_clickstream;
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: 0.219 seconds, Fetched: 14 row(s)
```

Checking data in both tables.

set hive.cli.print.header=true;

select * from clickstream_info limit 5 ;

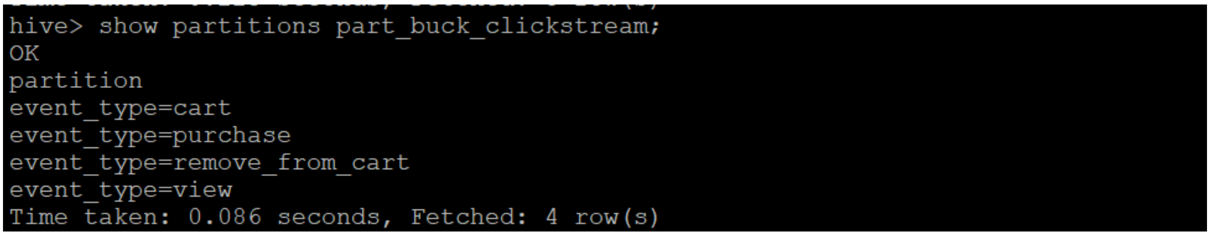
select * from part_buck_clickstream limit 5 ;



```
hive> set hive.cli.print.header=true;
hive> select * from clickstream_info limit 5 ;
OK
clickstream_info.event_time     clickstream_info.event_type     clickstream_info.product_id     clickstream_info.category_id    clickstream_info.category_cod
e       clickstream_info.brand  clickstream_info.price  clickstream_info.user_id        clickstream_info.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
Time taken: 0.284 seconds, Fetched: 5 row(s)
hive> select * from part_buck_clickstream limit 5 ;
OK
part_buck_clickstream.event_time        part_buck_clickstream.product_id        part_buck_clickstream.category_id       part_buck_clickstream.category_code p
art_buck_clickstream.brand      part_buck_clickstream.price     part_buck_clickstream.user_id   part_buck_clickstream.user_session      part_buck_clickstream
.event_type
2019-10-09 13:01:14 UTC 5863824 1487580005713052531             ingarden        4.44    462265274       af590a32-c73f-4833-8b9e-6683891e8df5    cart
2019-10-09 13:01:14 UTC 5883103 1487580005713052531             ingarden        4.44    462265274       af590a32-c73f-4833-8b9e-6683891e8df5    cart
2019-10-09 13:00:56 UTC 5883103 1487580005713052531             ingarden        4.44    462265274       af590a32-c73f-4833-8b9e-6683891e8df5    cart
2019-10-09 13:00:46 UTC 5871041 1487580005754995573                     4.92    558453153       ec6736cb-49df-4366-ab52-ec672b491928    cart
2019-10-09 11:20:30 UTC 5670323 1487580005754995573                     4.44    427884666       93ee8667-8f1e-4d16-b8f9-40fa7c7b4df4    cart
Time taken: 0.223 seconds, Fetched: 5 row(s)
```

Checking if partitions were created successfully.

show partitions part_buck_clickstream;



```
hive> show partitions part_buck_clickstream;
OK
partition
event_type=cart
event_type=purchase
event_type=remove_from_cart
event_type=view
Time taken: 0.086 seconds, Fetched: 4 row(s)
```

Overall, we have made two tables,

- One common table named clickstream_info which contains data of both October & November.
- One table with partitions & buckets named part_buck_clickstream for optimised querying which also contains data of both October & November.

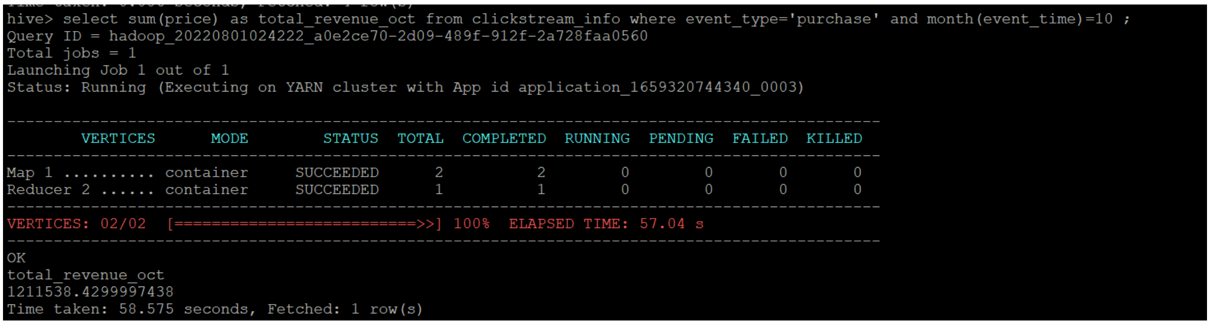
So, all the preparations are done & now we can move to query analysis-

<u>QUERY ANALYSIS</u>

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

a) Unoptimized query:

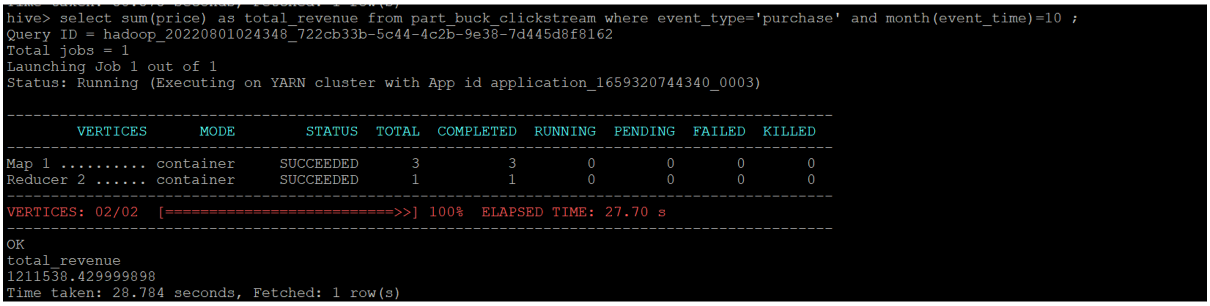
select sum(price) as total_revenue_oct from clickstream_info where event_type='purchase' and month(event_time)=10 ;



```
hive> select sum(price) as total_revenue_oct from clickstream_info where event_type='purchase' and month(event_time)=10 ;
Query ID = hadoop_20220801024222_a0e2ce70-2d09-489f-912f-2a728faa0560
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 57.04 s
----------------------------------------------------------------------------------------------
OK
total_revenue_oct
1211538.4299997438
Time taken: 58.575 seconds, Fetched: 1 row(s)
```

Optimized query:

select sum(price) as total_revenue from part_buck_clickstream where event_type='purchase' and month(event_time)=10 ;

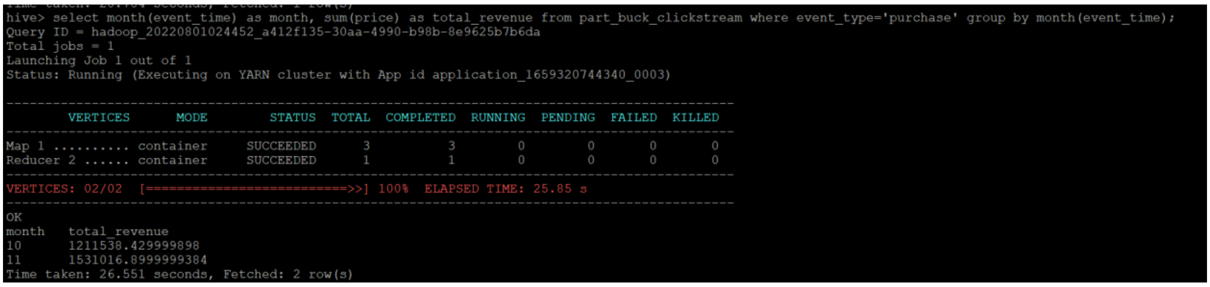


```
hive> select sum(price) as total_revenue from part_buck_clickstream where event_type='purchase' and month(event_time)=10 ;
Query ID = hadoop_20220801024348_722cb33b-5c44-4c2b-9e38-7d445d8f8162
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 27.70 s
----------------------------------------------------------------------------------------------
OK
total_revenue
1211538.429999898
Time taken: 28.784 seconds, Fetched: 1 row(s)
```

The total revenue generated in October is 1211538.429. Optimized query took 28.784 secs while unoptimized query took 57.04 secs to fetch the same result.

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

select month(event_time) as month, sum(price) as total_revenue from part_buck_clickstream where event_type='purchase' group by month(event_time);

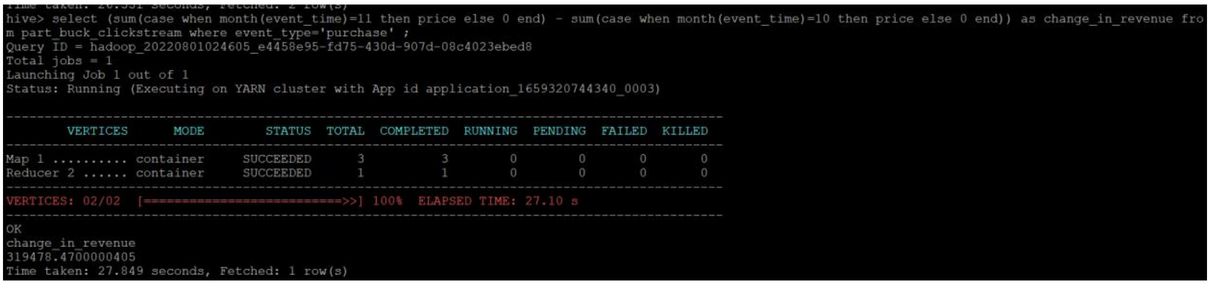


```
hive> select month(event_time) as month, sum(price) as total_revenue from part_buck_clickstream where event_type='purchase' group by month(event_time);
Query ID = hadoop_20220801024452_a412f135-30aa-4990-b98b-8e9625b7b6da
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 25.85 s
----------------------------------------------------------------------------------------------
OK
month   total_revenue
10      1211538.429999898
11      1531016.8999999384
Time taken: 26.551 seconds, Fetched: 2 row(s)
```

Total sum of purchases for October is 1211538.429 while for the November it’s 1531016.899.

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

select (sum(case when month(event_time)=11 then price else 0 end) - sum(case when month(event_time)=10 then price else 0 end)) as change_in_revenue from part_buck_clickstream where event_type='purchase' ;

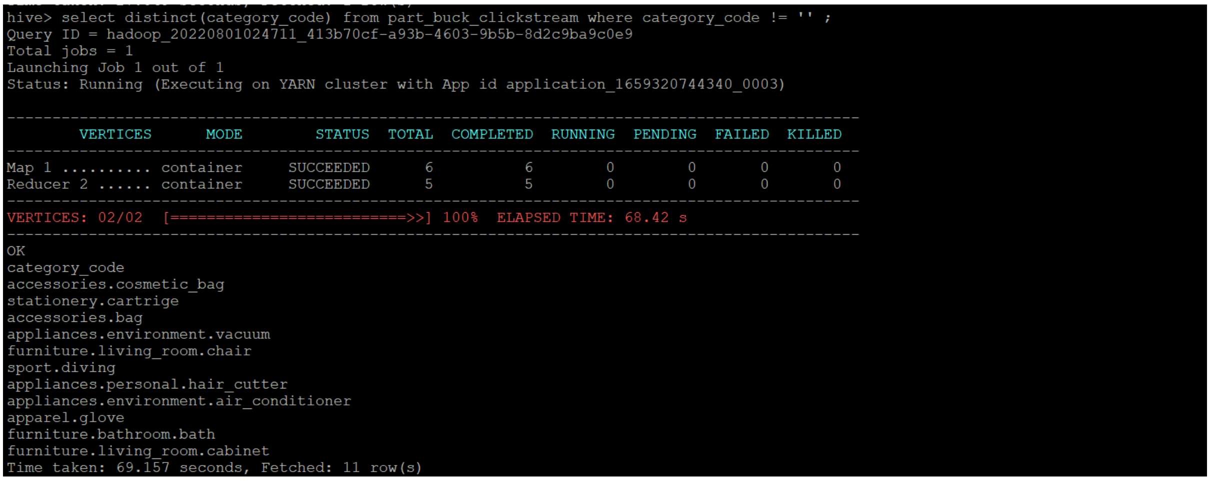


```
hive> select (sum(case when month(event_time)=11 then price else 0 end) - sum(case when month(event_time)=10 then price else 0 end)) as change_in_revenue fro
m part_buck_clickstream where event_type='purchase' ;
Query ID = hadoop_20220801024605_e4458e95-fd75-430d-907d-08c4023ebed8
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 27.10 s
----------------------------------------------------------------------------------------------
OK
change_in_revenue
319478.4700000405
Time taken: 27.849 seconds, Fetched: 1 row(s)
```

Change in revenue is 319478.47

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

select distinct(category_code) from part_buck_clickstream where category_code != '' ;

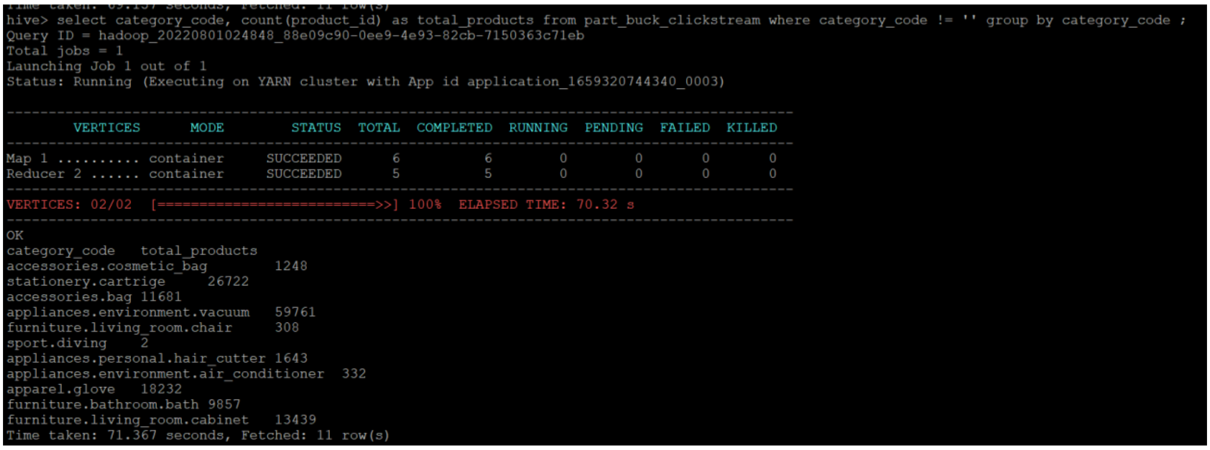


```
hive> select distinct(category_code) from part_buck_clickstream where category_code != '' ;
Query ID = hadoop_20220801024711_413b70cf-a93b-4603-9b5b-8d2c9ba9c0e9
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      5          5        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 68.42 s
----------------------------------------------------------------------------------------------
OK
category_code
accessories.cosmetic_bag
stationery.cartrige
accessories.bag
appliances.environment.vacuum
furniture.living_room.chair
sport.diving
appliances.personal.hair_cutter
appliances.environment.air_conditioner
apparel.glove
furniture.bathroom.bath
furniture.living_room.cabinet
Time taken: 69.157 seconds, Fetched: 11 row(s)
```

There are 6 categories and 11 distinct sub-categories

5. Find the total number of products available under each category.

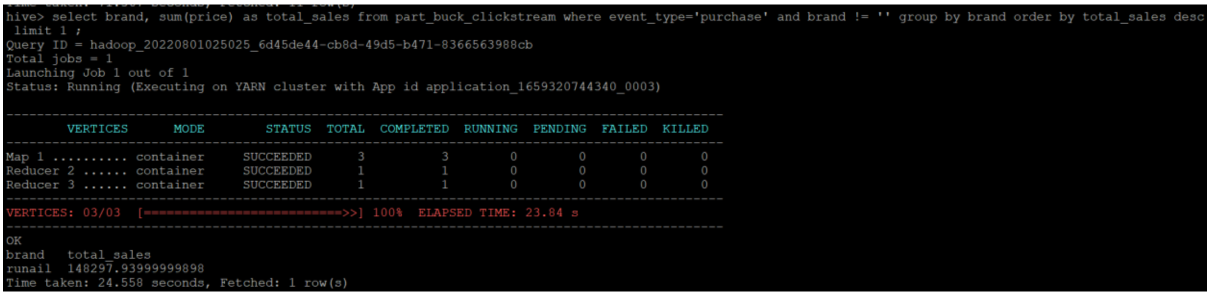
select category_code, count(product_id) as total_products from part_buck_clickstream where category_code != '' group by category_code ;



```
hive> select category_code, count(product_id) as total_products from part_buck_clickstream where category_code != '' group by category_code ;
Query ID = hadoop_20220801024848_88e09c90-0ee9-4e93-82cb-7150363c71eb
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      6          6        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      5          5        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 70.32 s
----------------------------------------------------------------------------------------------
OK
category_code   total_products
accessories.cosmetic_bag        1248
stationery.cartrige     26722
accessories.bag 11681
appliances.environment.vacuum   59761
furniture.living_room.chair     308
sport.diving    2
appliances.personal.hair_cutter 1643
appliances.environment.air_conditioner  332
apparel.glove   18232
furniture.bathroom.bath 9857
furniture.living_room.cabinet   13439
Time taken: 71.367 seconds, Fetched: 11 row(s)
```

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

select brand, sum(price) as total_sales from part_buck_clickstream where event_type='purchase' and brand != '' group by brand order by total_sales desc limit 1 ;

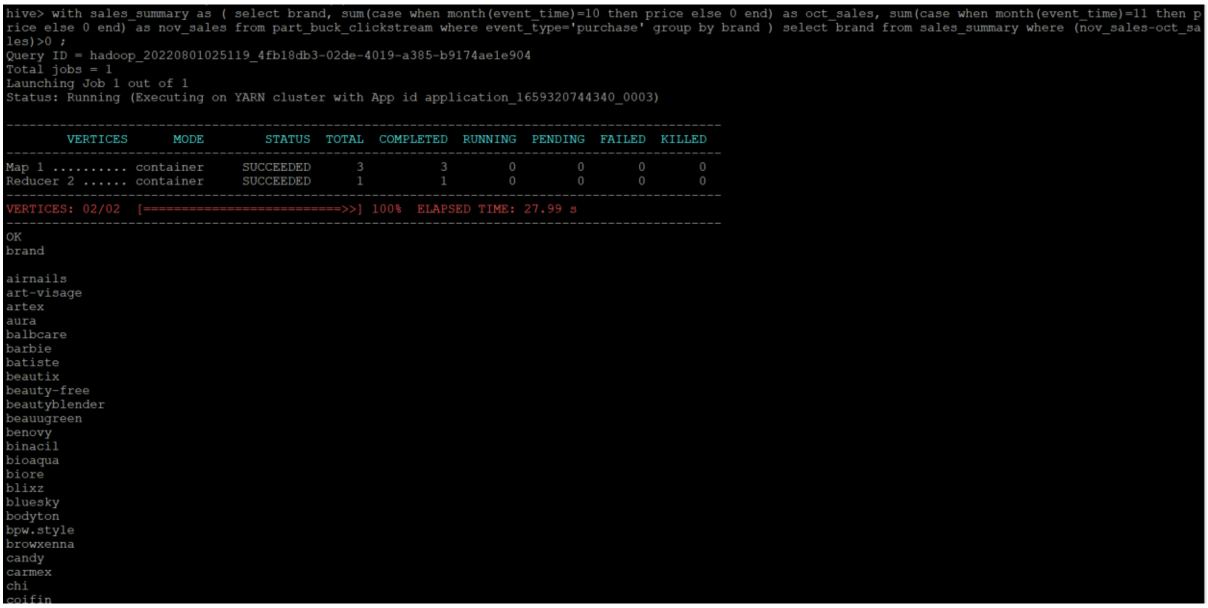


```
hive> select brand, sum(price) as total_sales from part_buck_clickstream where event_type='purchase' and brand != '' group by brand order by total_sales desc
 limit 1 ;
Query ID = hadoop_20220801025025_6d45de44-cb8d-49d5-b471-8366563988cb
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 23.84 s
----------------------------------------------------------------------------------------------
OK
brand   total_sales
runail  148297.93999999898
Time taken: 24.558 seconds, Fetched: 1 row(s)
```

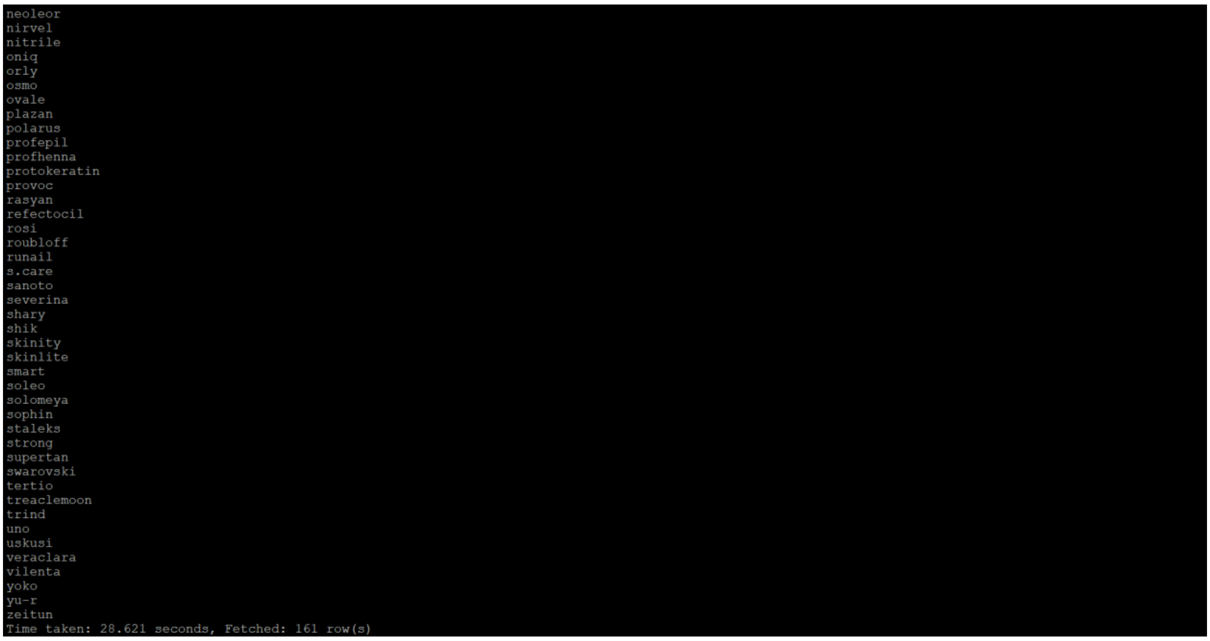
The top brand is runail with total_sales 148297.939

7. Which brands increases their sales from October to November?

with sales_summary as (select brand, sum(case when month(event_time)=10 then price else 0 end) as oct_sales, sum(case when month(event_time)=11 then price else 0 end) as nov_sales from part_buck_clickstream where event_type='purchase' group by brand) select brand from sales_summary where (nov_sales-oct_sales)>0 ;



```
hive> with sales_summary as ( select brand, sum(case when month(event_time)=10 then price else 0 end) as oct_sales, sum(case when month(event_time)=11 then p
rice else 0 end) as nov_sales from part_buck_clickstream where event_type='purchase' group by brand ) select brand from sales_summary where (nov_sales-oct_sa
les)>0 ;
Query ID = hadoop_20220801025119_4fb18db3-02de-4019-a385-b9174ae1e904
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 27.99 s
----------------------------------------------------------------------------------------------
OK
brand

airnails
art-visage
artex
aura
balbcare
barbie
batiste
beautix
beauty-free
beautyblender
beauugreen
benovy
binacil
bioaqua
biore
blixz
bluesky
bodyton
bpw.style
browxenna
candy
carmex
chi
coifin
```

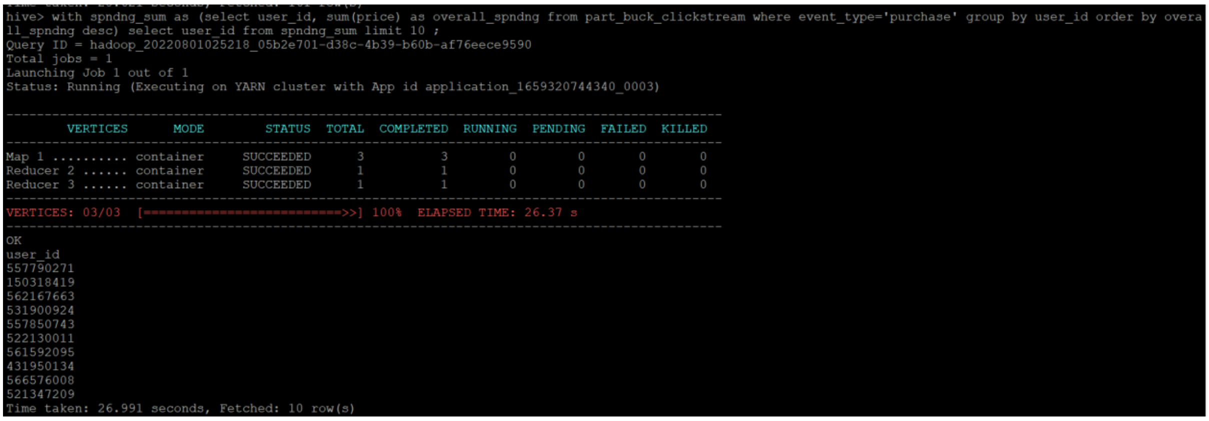


```
neoleor
nirvel
nitrile
oniq
orly
osmo
ovale
plazan
polarus
profepil
profhenna
protokeratin
provoc
rasyan
refectocil
rosi
roubloff
runail
s.care
sanoto
severina
shary
shik
skinity
skinlite
smart
soleo
solomeya
sophin
staleks
strong
supertan
swarovski
tertio
treaclemoon
trind
uno
uskusi
veraclara
vilenta
yoko
yu-r
zeitun
Time taken: 28.621 seconds, Fetched: 161 row(s)
```

There is a total of 161 brands that had increased sales from October to November.

8. Your company wants to reward the top 10 users of its website with a Golden Customer plan. Write a query to generate a list of top 10 users who spend the most.

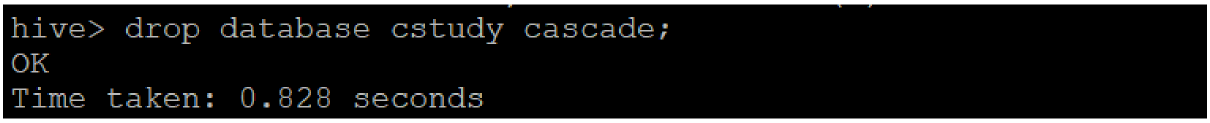
with spndng_sum as (select user_id, sum(price) as overall_spndng from part_buck_clickstream where event_type='purchase' group by user_id order by overall_spndng desc) select user_id from spndng_sum limit 10 ;



```
hive> with spndng_sum as (select user_id, sum(price) as overall_spndng from part_buck_clickstream where event_type='purchase' group by user_id order by overa
ll_spndng desc) select user_id from spndng_sum limit 10 ;
Query ID = hadoop_20220801025218_05b2e701-d38c-4b39-b60b-af76eece9590
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1659320744340_0003)

----------------------------------------------------------------------------------------------
        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: 26.37 s
----------------------------------------------------------------------------------------------
OK
user_id
557790271
150318419
562167663
531900924
557850743
522130011
561592095
431950134
566576008
521347209
Time taken: 26.991 seconds, Fetched: 10 row(s)
```

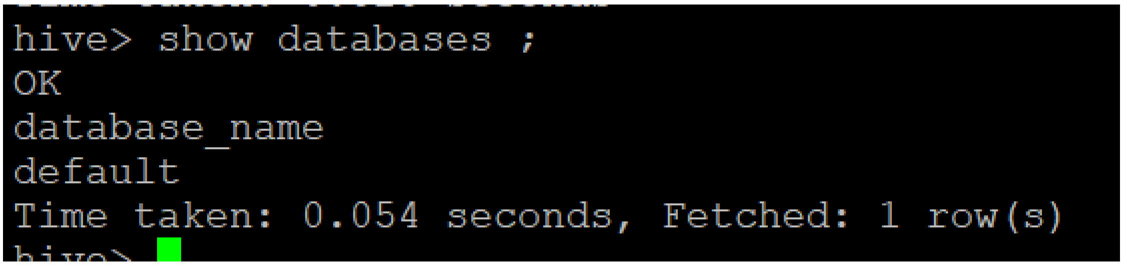
Dropping database

drop database cstudy cascade;



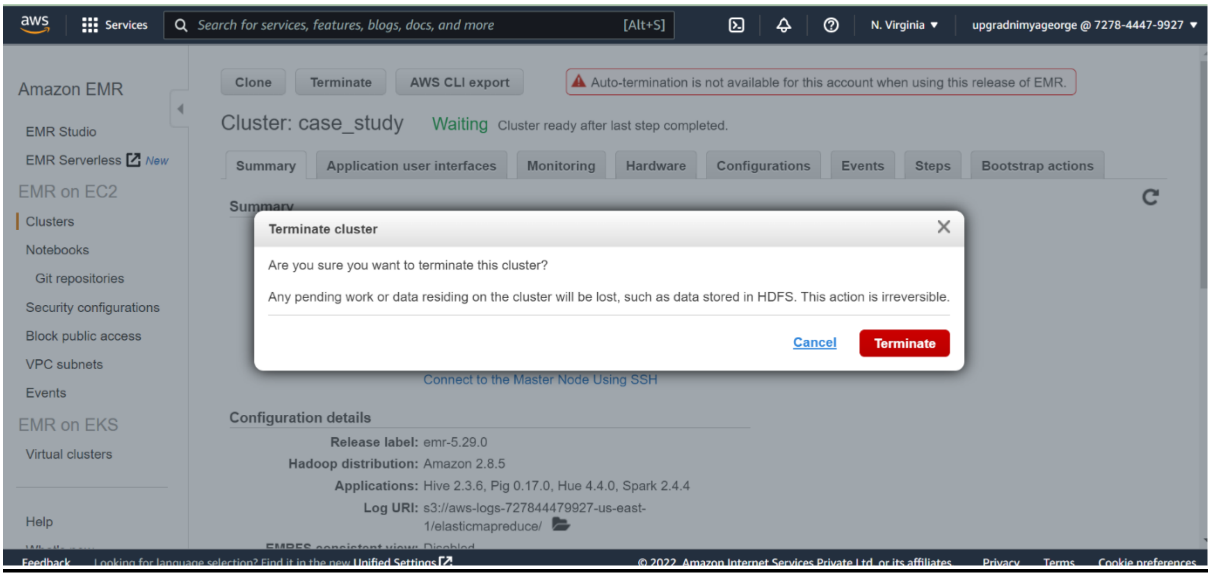
```
hive> drop database cstudy cascade;
OK
Time taken: 0.828 seconds
```

show databases;



```
hive> show databases ;
OK
database_name
default
Time taken: 0.054 seconds, Fetched: 1 row(s)
hive>
```

TERMINATING THE EMR CLUSTER



Auto-termination is not available for this account when using this release of EMR.

Cluster: case_study Waiting Cluster ready after last step completed.

Terminate cluster

Are you sure you want to terminate this cluster?

Any pending work or data residing on the cluster will be lost, such as data stored in HDFS. This action is irreversible.

Configuration details

Release label: emr-5.29.0
Hadoop distribution: Amazon 2.8.5
Applications: Hive 2.3.6, Pig 0.17.0, Hue 4.4.0, Spark 2.4.4
Log URI: s3://aws-logs-727844479927-us-east-1/elasticmapreduce/
EMRFS consistent view: Disabled

On clicking terminate the cluster gets terminated and then we can log out from our AWS console.