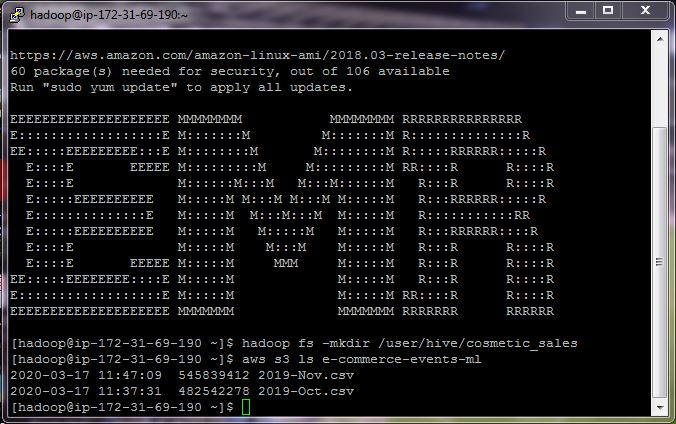
**UPGRAD HIVE CASE STUDY**

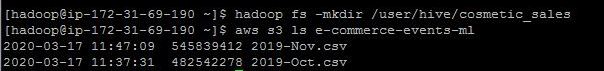
The following steps are performed in the hive case study :

1. Connect the local machine to the master node using SSH



1. Create a folder name ‘Cosmetic\_Sales’ in the HDFS using the following command:

***hadoop fs -mkdir /user/hive/cosmetic\_sales***

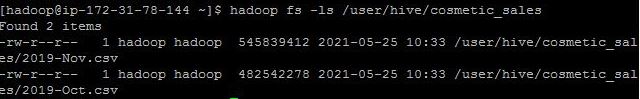
******

1. Import the data to the folder ‘cosmetic\_sales’ in the HDFS using the following command:

***hadoop distcp s3://e-commerce-events-ml/ /user/hive/cosmetic\_sales/***

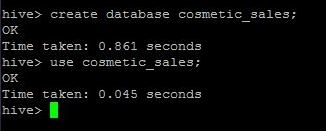
1. Now we have imported the data in the HDFS. To see the imported data run the following command :

***Hadoop fs-ls /user/hive/cosmetic\_sales***



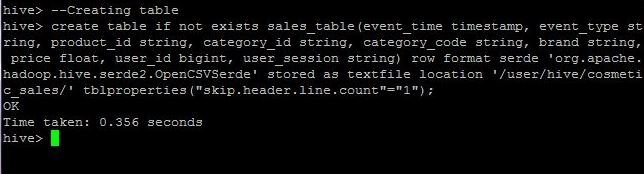
**Here,we can see that both the files are uploaded in the HDFS**

1. Launch the Hive Service .For this run the command “***hive”***
2. Creating and using the database named ‘cosmetic\_sales’ using the following query :



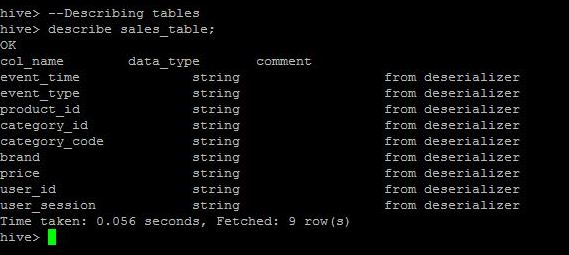
1. Create the external table by using the following query:

***create table if not exists sales\_table(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' stored as textfile location '/user/hive/cosmetic\_sales/' tblproperties("skip.header.line.count"="1");***

******

1. Describe the table ‘sales\_table’ by using the following query :

***Describe sales\_table ;***

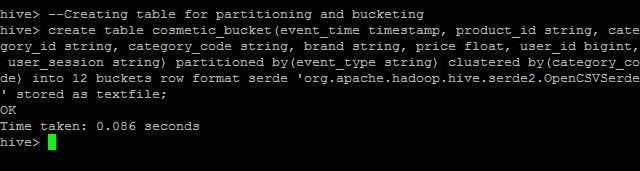


1. To show the headers for all the queries use the following query :

***set hive.cli.print.header=true ;***

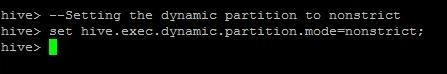
1. Create the partitioning and bucketing using the following command :

***create table cosmetic\_bucket(event\_time timestamp, product\_id string, category\_id string, category\_code string, brand string, price float, user\_id bigint, user\_session string) partitioned by(event\_type string) clustered by(category\_code) into 12 buckets row format serde 'org.apache.hadoop.hive.serde2.OpenCSVSerde' stored as textfile;***

******

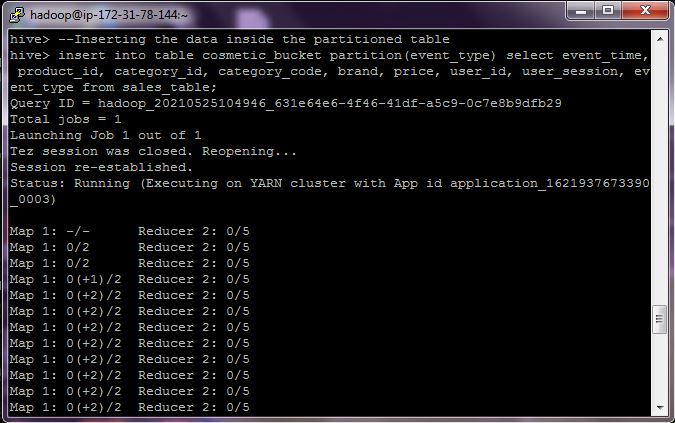
11) Set dynamic partitioning mode to nonstrict using the following command :

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



1. Load the data in the partitioned and bucketed table named ‘ext\_table2019’ using the following command :

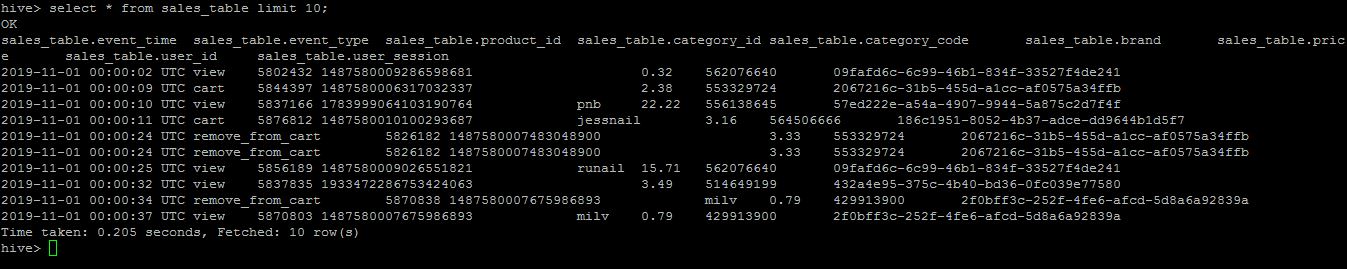
***insert into table cosmetic\_bucket partition(event\_type) select event\_time, product\_id, category\_id, category\_code, brand, price, user\_id, user\_session, event\_type from sales\_table;***



# QUERY OPTIMIZATION

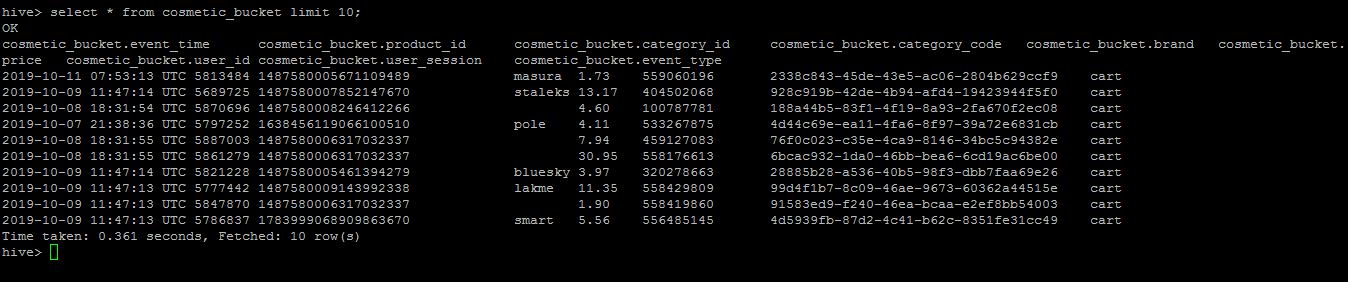
We have created the partitioned and bucketed table named ‘cosmetic\_bucket’ to optimize the queries. Let’s see how we can optimize queries through the example.

1. Fetching the first 10 rows of ***sales\_table.***



Here we can see that the time taken to execute the query in the table ‘sales\_table’ is 0.205 seconds.

1. Fetching the first 10 rows of ***cosmetic\_bucket.***



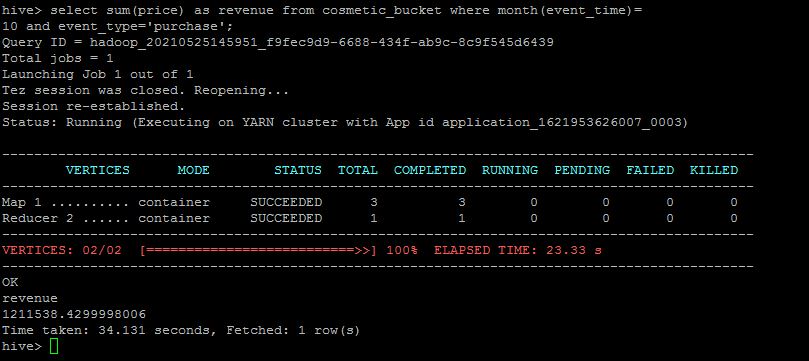
Here we can see that the time taken to execute the query in the table ‘cosmetic\_bucket’ is 0.361 seconds.

**CONCLUSION:** We can conclude that the partitioned and bucketed table ‘cosmetic\_bucket’ takes less time as compared to table ‘sales\_table’ to execute the query**.** So now onwards, we will perform all the queries on the partitioned and bucketed table ‘cosmetic\_table’.

# QUESTIONS AND ANSWERS

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

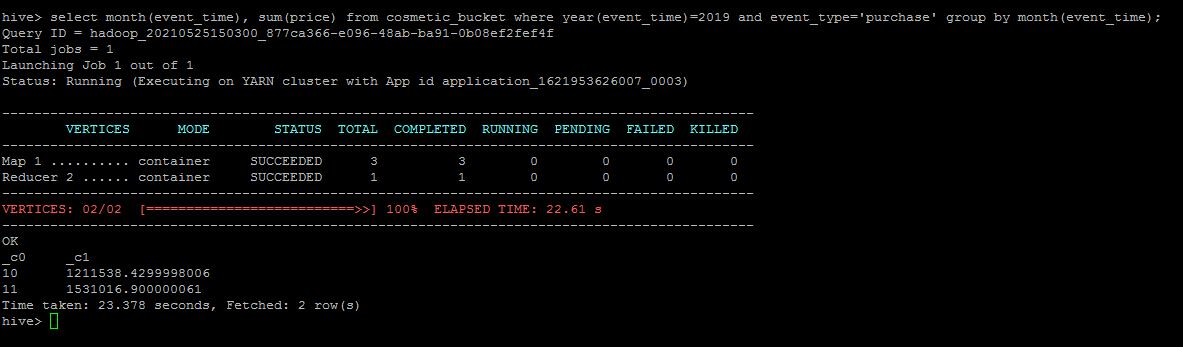
Ans: ***select sum(price) as revenue from cosmetic\_bucket where month(event\_time)=10 and event\_type='purchase';***



The total revenue generated due to the purchases made in October month is **1211538.4299998006 .**

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

Ans ***select month(event\_time), sum(price) from cosmetic\_bucket where year(event\_time)=2019 and event\_type='purchase' group by month(event\_time);***

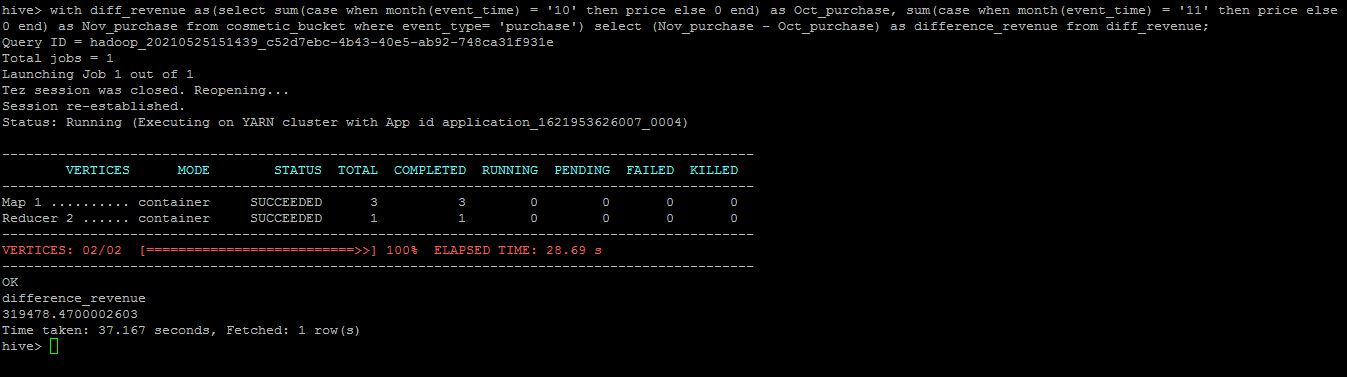
******

The total sum of purchases in the month of October is **1211538.4299998006** and the total sum of purchases in the month of November is **1531016.900000061 .**

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

**ANS:**

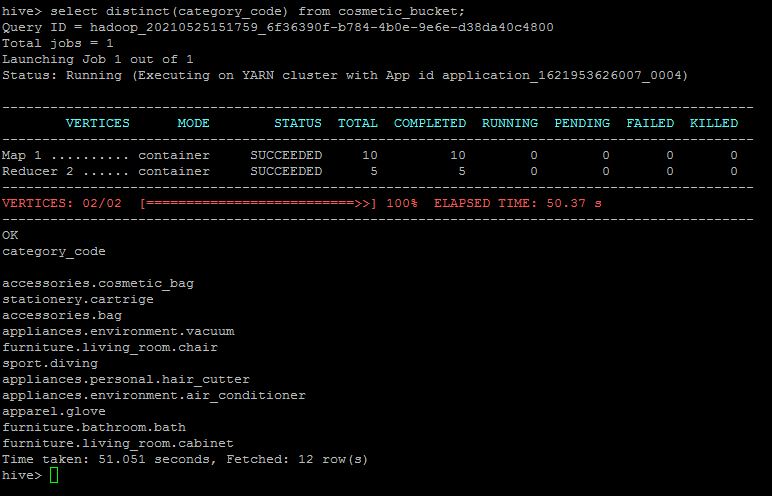
***with diff\_revenue as(select sum(case when month(event\_time) = '10' then price else 0 end) as Oct\_purchase, sum(case when month(event\_time) = '11' then price else 0 end) as Nov\_purchase from cosmetic\_bucket where event\_type= 'purchase') select (Nov\_purchase - Oct\_purchase) as difference\_revenue from diff\_revenue;***

******

The change in revenue generated due to purchases from October to November is **319478.4700002603.**

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

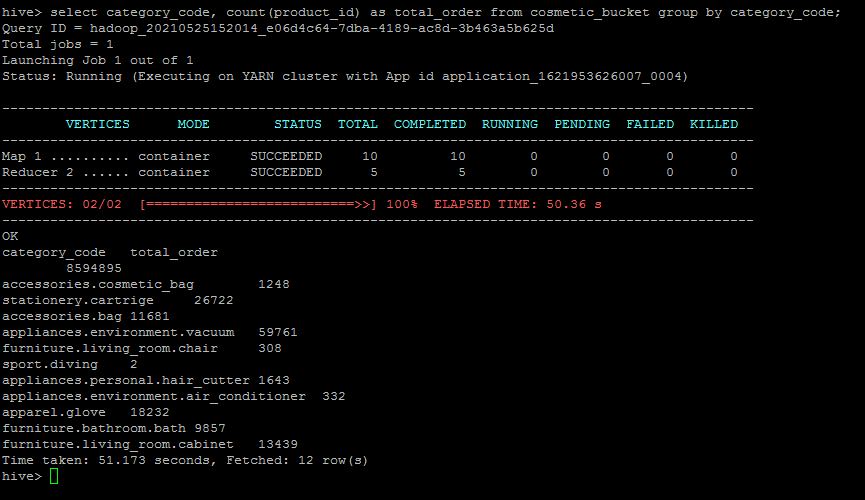
ANS: ***select distinct(category\_code) from cosmetic\_bucket;***

******

The distinct categories of the products are as follows :

1. ***Accessories.cosmetic\_bag***
2. ***Stationary.cartrige***
3. ***Accessories.bag***
4. ***Appliances.environment.vacuum***
5. ***Furniture.living\_room.chair***
6. ***Sport.diving***
7. ***Appliances.person.hair\_cutter***
8. ***Appliances.environment.air\_conditioner***
9. ***Apparel.glove***
10. ***Furniture.bathroom.bath***
11. ***Furniture.living\_room.cabinet***
12. **Find the total number of products available under each category.**

***select category\_code, count(product\_id) as total\_order from cosmetic\_bucket group by category\_code;***

******

The total number of products available under each category is as follows

***: i) Accessories.cosmetic\_bag - 1248***

***ii) Stationary.cartrige - 26722***

***iii) Accessories.bag – 11681***

***iv) Appliances.environment.vacuum – 59761***

***v) Furniture.living\_room.chair - 308***

***vi) Sport.diving - 2***

***vii) Appliances.person.hair\_cutter -1643***

***viii) Appliances.environment.air\_conditioner - 332***

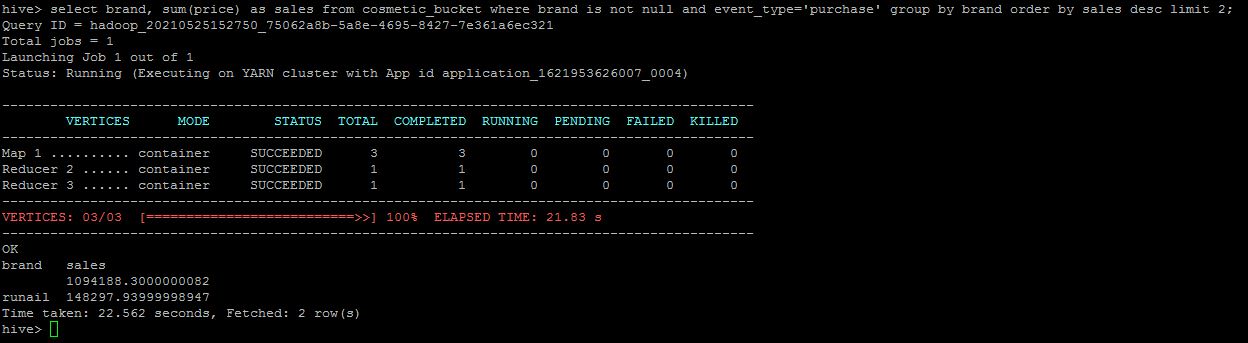
***ix) Apparel.glove - 18232***

***x) Furniture.bathroom.bath - 9857***

***xi) Furniture.living\_room.cabinet – 13439***

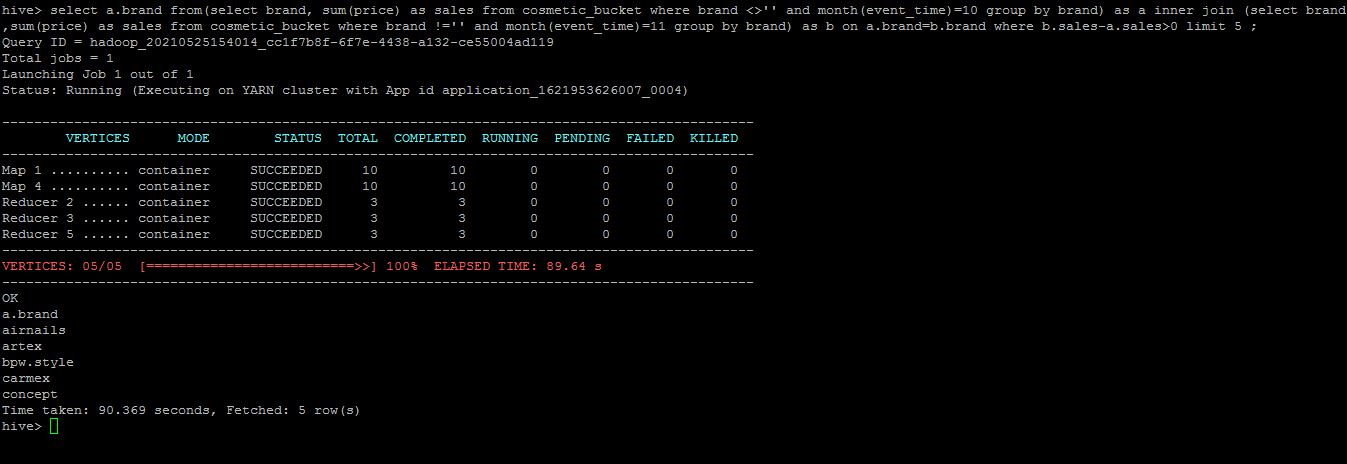
1. **Which brand had the maximum sales in October and November combined?**

***select brand, sum(price) as sales from cosmetic\_bucket where brand is not null and event\_type='purchase' group by brand order by sales desc limit 2;***

******

1. **Which brands increased their sales from October to November?**

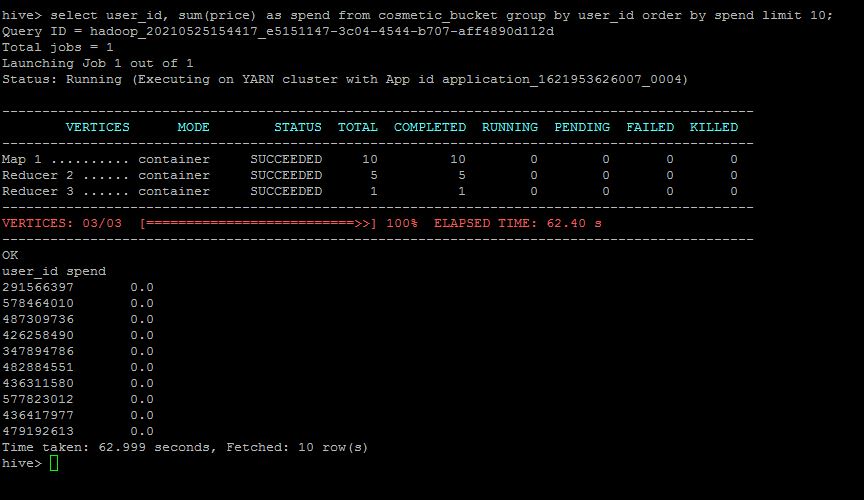
***select a.brand from(select brand, sum(price) as sales from cosmetic\_bucket where brand <>'' and month(event\_time)=10 group by brand) as a inner join (select brand,sum(price) as sales from cosmetic\_bucket where brand !='' and month(event\_time)=11 group by brand) as b on a.brand=b.brand where b.sales-a.sales>0 limit 5 ;***

******

The following top 5 brands increased their sales from October to November:

1. ***airnails***
2. ***artex***
3. ***bpw.style***
4. ***carmex***
5. ***concept***
6. **Your company wants to reward the top 10 users of its website with a Golden Customer plan. Write a query to generate a list of top 10 users who spend the most.**

***select user\_id, sum(price) as spend from cosmetic\_bucket group by user\_id order by spend limit 10;***

******