

DATA INGESGTION AND PIPELINE ANALYSIS WITHIN MySQL, HDFS AND HIVE.

COURSE: BDM 1024

Group Members

Mahima Akula

Modupeola Omodunni Oyatokun

Jumoke Yekeen

Chibuike Okoroama

Diksha

Harish Kundal

SUBMITTED ON 29-07-2023

TABLE OF CONTENTS:

1. Introduction
 - 1.1. Project Overview
 - 1.2. Data Ingestion and Preparation
2. Data Transfer:
 - 2.1. Data transfer to MySQL
 - 2.2. Creating Database in MySQL
 - 2.3. Results of uploading data into MySQL
3. Exporting data from MySQL to HDFS using Sqoop
 - 3.1. Transferring Data from MySQL to HDFS using Sqoop
 - 3.2. Data ingestion into Hive
4. Data Analysis using Hive
 - 4.1. Exporting Hive Query Results to Hive Result Tables
 - 4.2. Exporting data from Hive to MySQL
5. Data Insights
6. Conclusion
7. Challenges Faced
8. Project Participants

1. INTRODUCTION:

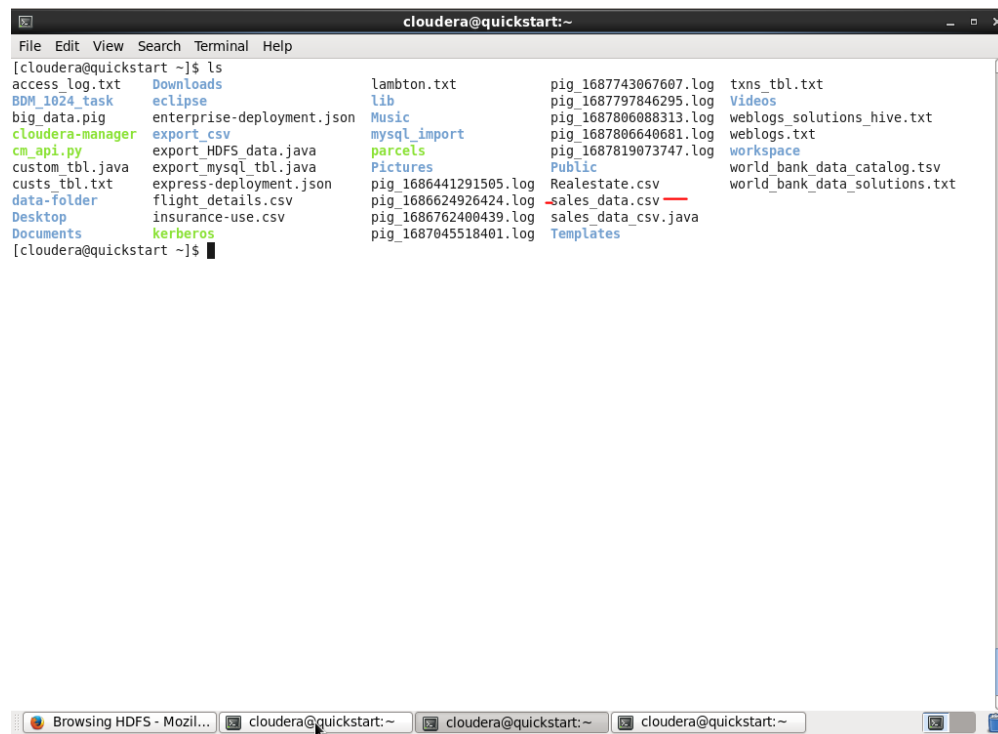
1.1.PROJECT OVERVIEW:

For this project (BDM 1024 project), sales_data.csv file was cleaned, data type formats were set, additional unique ID column was added, and we made sure that the data is in .csv format before we ingested into cloudera. It was important that we do not tamper with the data so that meaningful insights are not lost as result of such action.

1.2. DATA INGESTION AND PREPARATION:

The challenge was to ingest a .csv file into cloudera local drive, analyze it in Hive and store the analyzed results in MySQL. It is expected that a minimum of five queries to analyze the data is expected. The data to be ingested into cloudera was in the local drive of computer, therefore we needed to transfer it into cloudera.

To transfer the data into the HDFS, we used FileZilla to connect to cloudera local drive and then transfer the file into cloudera local drive. Figure 1 below shows sales_data.csv located in the local cloudera drive. By simply using the Linux command `ls` we visualized different files including the file that will be analyzed.

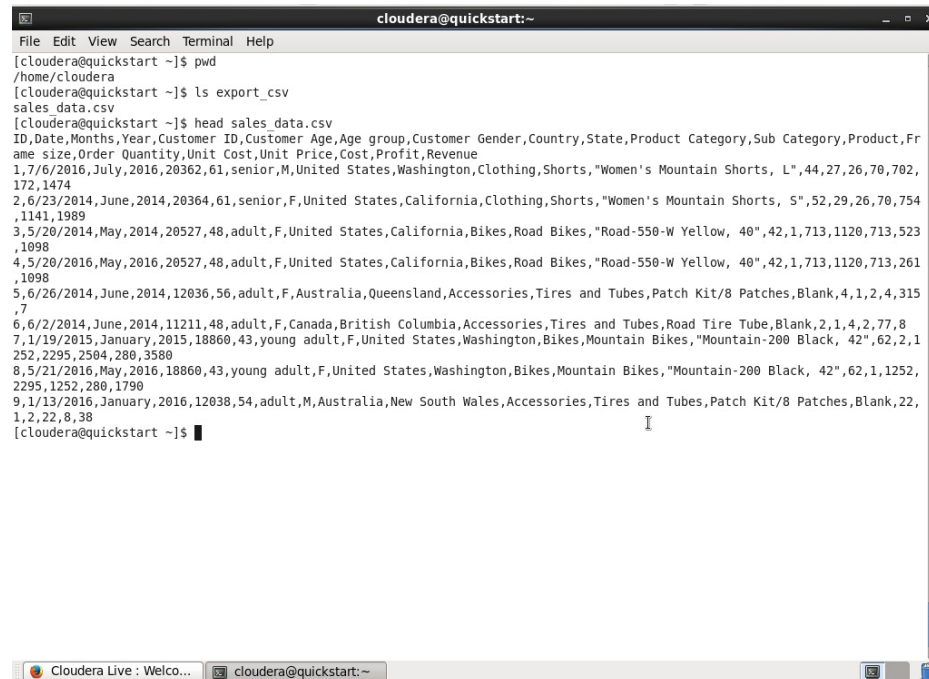


```
cloudera@quickstart:~$ ls
access_log.txt      Downloads          lambton.txt        pig_1687743067607.log  txns_tbl.txt
BDM_1024_task      eclipse           lib                pig_1687797846295.log  Videos
big_data.pig        enterprise-deployment.json  Music              pig_1687806088313.log  weblogs_solutions_hive.txt
cloudera-manager    export_csv        mysql_import       pig_1687806640681.log  weblogs.txt
cm_api.py           export_HDFS_data.java  parcels            pig_1687819073747.log  workspace
custom_tbl.java     export_mysql_tbl.java  Pictures           Public                 world_bank_data_catalog.tsv
custs_tbl.txt       express-deployment.json  pig_1686441291505.log  Realestate.csv         world_bank_data_solutions.txt
data-folder         flight_details.csv     pig_1686624926424.log  -sales_data.csv
Desktop             insurance-use.csv      pig_1686762400439.log  sales_data_csv.java
Documents           kerberos             pig_1687045518401.log  Templates
```

Figure 1. show sale_data.csv in the cloudera local drive

Now that we have our data, sales_data.csv in the local drive, next thing is to display the data in the terminal. In our project we used `head -n 5` to display few results but we wanted to see the entire

data. Figure 2 below shows a display of sale_data.csv data content. This way we confirmed that the right data was transferred. However, we could not do any further analysis on the data than display it on the terminal. Therefore, the challenge of analyzing the data in Hive and storing it in MySQL persists.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ pwd  
/home/cloudera  
[cloudera@quickstart ~]$ ls export_csv  
sales_data.csv  
[cloudera@quickstart ~]$ head sales_data.csv  
ID,Date,Months,Year,Customer ID,Customer Age,Age group,Customer Gender,Country,State,Product Category,Sub Category,Product,Frame size,Order Quantity,Unit Cost,Unit Price,Cost,Profit,Revenue  
1,7/6/2016,July,2016,20362,61,senior,M,United States,Washington,Clothing,Shorts,"Women's Mountain Shorts, L",44,27,26,70,702,172,1474  
2,6/23/2014,June,2014,20364,61,senior,F,United States,California,Clothing,Shorts,"Women's Mountain Shorts, S",52,29,26,70,754,1141,1989  
3,5/20/2014,May,2014,20527,48,adult,F,United States,California,Bikes,Road Bikes,"Road-550-W Yellow, 40",42,1,713,1120,713,523,1098  
4,5/20/2016,May,2016,20527,48,adult,F,United States,California,Bikes,Road Bikes,"Road-550-W Yellow, 40",42,1,713,1120,713,261,1098  
5,6/26/2014,June,2014,12036,56,adult,F,Australia,Queensland,Accessories,Tires and Tubes,Patch Kit/8 Patches,Blank,4,1,2,4,315,7  
6,6/2/2014,June,2014,11211,48,adult,F,Canada,British Columbia,Accessories,Tires and Tubes,Road Tire Tube,Blank,2,1,4,2,77,8  
7,1/19/2015,January,2015,18860,43,young adult,F,United States,Washington,Bikes,Mountain Bikes,"Mountain-200 Black, 42",62,2,1252,2295,2504,280,3580  
8,5/21/2016,May,2016,18860,43,young adult,F,United States,Washington,Bikes,Mountain Bikes,"Mountain-200 Black, 42",62,1,1252,2295,1252,280,1790  
9,1/13/2016,January,2016,12038,54,adult,M,Australia,New South Wales,Accessories,Tires and Tubes,Patch Kit/8 Patches,Blank,22,1,2,22,8,38  
[cloudera@quickstart ~]$
```

Figure 2 shows display of sale_data.csv on the terminal

2. DATA TRANSFER:

2.1. CREATING DATABASE IN MySQL:

According to the project task, it is mandatory that we ingest the sale_data.csv into a table in MySQL database in cloudera, therefore we had to confirm that there was no replica table in the MySQL database. Figure 3 shows that we log into MySQL with error in cloudera by typing MySQL -u root -p in the cloudera terminal. With MySQL syntax show databases, figure 4 shows the different databases present in my MySQL database. On this note we must create a new database with a new table that is with the same schema as the sale_data.csv file before we can ingest it into the MySQL database.

2.2. CREATING TABLE IN MySQL:

Figure 4 shows the creation of a database named project_1024 in MySQL. A table with similar .csv schema was created, then described to confirm that the schema matches properly. As mentioned in introduction, the ID column added to the .csv data was done to provide uniqueness to the data. So, the ID was made the primary key instead.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 14  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| cm |  
| custom |  
| firehose |  
| hue |  
| metastore |  
| mysql |  
| nav |  
| navms |  
| oozie |  
| retail_db |  
| rman |  
| sentry |  
| sqoop_export |  
+-----+  
14 rows in set (0.11 sec)  
  
mysql> create database project_1024;  
Query OK, 1 row affected (0.06 sec)  
  
mysql> show databases;
```

Figure 3.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
  
mysql> create table sales_csv (ID int PRIMARY KEY, date string, months varchar(11), Year year, cust_ID int, cust_Age int, age_group varchar(15), cust_gender char(5), country varchar(20), state varchar(25), prod_cat varchar(20), sub_cat varchar(20), prod varchar(50), frame_size int, order_qty int, unit_cost int, unit_price int, cost int, profit int, revenue int);  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'string, months varchar(11), Year year, cust_ID int, cust_Age int, age_group varc' at line 1  
mysql> create table sales_csv (ID int PRIMARY KEY, date varchar(15), months varchar(11), Year year, cust_ID int, cust_Age int, age_group varchar(15), cust_gender char(5), country varchar(20), state varchar(25), prod_cat varchar(20), sub_cat varchar(20), prod varchar(50), frame_size int, order_qty int, unit_cost int, unit_price int, cost int, profit int, revenue int);  
Query OK, 0 rows affected (0.11 sec)  
  
mysql> describe sales_csv;  
+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+  
| ID | int(11) | NO | PRI | NULL | |  
| date | varchar(15) | YES | | NULL | |  
| months | varchar(11) | YES | | NULL | |  
| Year | year(4) | YES | | NULL | |  
| cust_ID | int(11) | YES | | NULL | |  
| cust_Age | int(11) | YES | | NULL | |  
| age_group | varchar(15) | YES | | NULL | |  
| cust_gender | char(5) | YES | | NULL | |  
| country | varchar(20) | YES | | NULL | |  
| state | varchar(25) | YES | | NULL | |  
| prod_cat | varchar(20) | YES | | NULL | |  
| sub_cat | varchar(20) | YES | | NULL | |  
| prod | varchar(50) | YES | | NULL | |  
| frame_size | int(11) | YES | | NULL | |  
| order_qty | int(11) | YES | | NULL | |  
| unit_cost | int(11) | YES | | NULL | |  
| unit_price | int(11) | YES | | NULL | |  
| cost | int(11) | YES | | NULL | |  
| profit | int(11) | YES | | NULL | |  
| revenue | int(11) | YES | | NULL | |  
+-----+  
20 rows in set (0.03 sec)  
  
mysql>
```

Figure 4 shows MySQL table create and schema description.

After creating the sale_csv table in MySQL, we ingested the sale_data.csv date into the table from the MySQL terminal simply by using the load data infile location of data syntax as shown in figure 5. A total of 113036 rows were imported which is the same as .csv.

2.3. RESULTS OF UPLOADING DATA INTO MYSQL:

Therefore, our data is completely ingested into MySQL sales_csv table in project_1024 database. With the select * from the sales_csv table we can see that our data has been successfully ingested.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
+-----+
| sub_cat | varchar(20) | YES | NULL |
| prod    | varchar(50) | YES | NULL |
| frame_size | int(11)    | YES | NULL |
| order_qty | int(11)    | YES | NULL |
| unit_cost | int(11)    | YES | NULL |
| unit_price | int(11)    | YES | NULL |
| cost     | int(11)    | YES | NULL |
| profit   | int(11)    | YES | NULL |
| revenue  | int(11)    | YES | NULL |
+-----+
20 rows in set (0.03 sec)

mysql> load data infile '/home/cloudera/sales_data.csv' into table sales_csv fields terminated by ',' enclosed by '"' lines terminated by '\n' ignore 1 lines;
Query OK, 113036 rows affected, 65535 warnings (2.02 sec)
Records: 113036 Deleted: 0 Skipped: 0 Warnings: 0

mysql> select * from sales_csv limit 5;
+-----+
| ID | date       | months | Year | cust_ID | cust_Age | age_group | cust_gender | country      | state      | prod_cat | sub_cat |
+-----+
| 1  | 7/6/2016   | July    | 2016 | 20362   | 61       | senior    | M           | United States | Washington | Clothing | Shorts |
| 2  | 6/23/2014  | June    | 2014 | 20364   | 61       | senior    | F           | United States | California | Clothing | Shorts |
| 3  | 5/20/2014  | May     | 2014 | 20527   | 48       | adult     | F           | United States | California | Bikes    | Road Bikes |
| 4  | 5/20/2016  | May     | 2016 | 20527   | 48       | adult     | F           | United States | California | Bikes    | Road Bikes |
| 5  | 6/26/2014  | June    | 2014 | 12036   | 56       | adult     | F           | Australia    | Queensland | Accessories | Tires and Tubes |
+-----+
5 rows in set (0.07 sec)

```

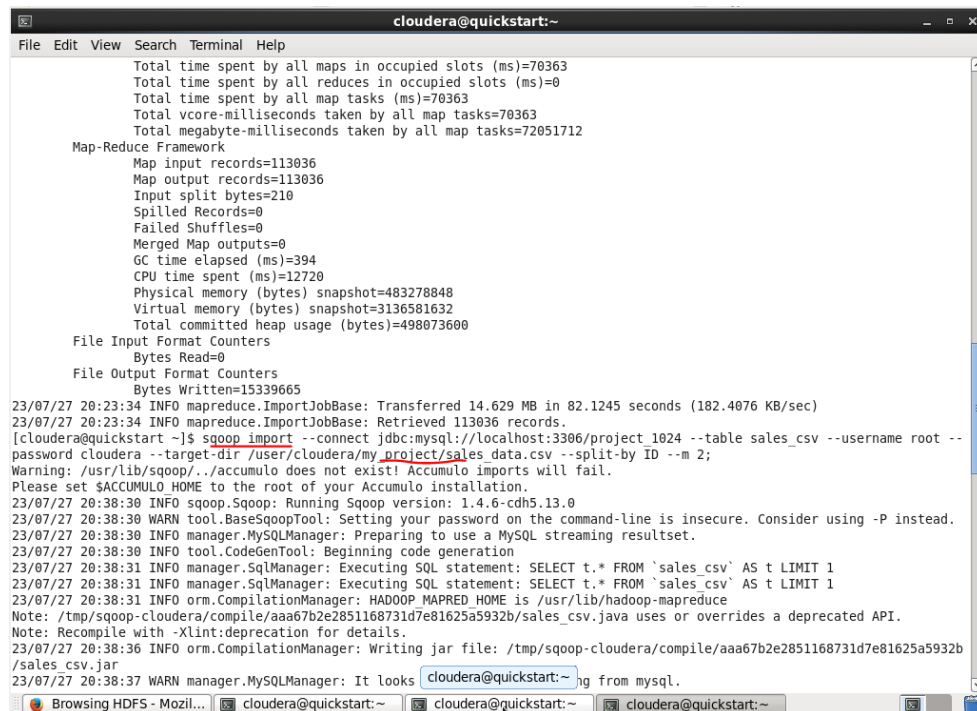
Figure 5 shows ingestion of csv file into MySQL and display of data.

3. EXPORTING DATA FROM MySQL TO HDFS USING SQOOP:

The uniqueness of this project is that we can move that between frameworks in Hadoop. However, the challenge of analyzing our data in Hive persists. To resolve this challenge, we must import our data into HDFS in cloudera then we can ingest the data into Hive. To achieve this, we must make sure make sure that we can transfer the entire MySQL table into HDFS, and we have created a database and exact same table schema in Hive for data ingestion.

3.1. TRANSFERRING DATA FROM MYSQL TO HDFS USING SQOOP:

Sqoop, which is a tool for importing and exporting bulk data between Hadoop and external data storages, we used it to transfer sales_csv into HDFS by using the Sqoop import –connect as shown in figure 6 below. Common mistakes made is to run this syntax on the same terminal as MySQL, however it is important we do it on a different cloudera HDFS terminal other than the MySQL terminal. The Sqoop import was split into two with the use ID in the data, which was made primary key, as highlighted in the syntax in figure 6, the data was successfully ingested into the HDFS. From HDFS, we can transfer the data into Hive for analysis. The ability to move data between framework HDFS and HDFS is why Sqoop is important for our structured data.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Total time spent by all maps in occupied slots (ms)=70363  
Total time spent by all reduces in occupied slots (ms)=0  
Total time spent by all map tasks (ms)=70363  
Total vcore-milliseconds taken by all map tasks=70363  
Total megabyte-milliseconds taken by all map tasks=72051712  
Map-Reduce Framework  
  Map input records=113036  
  Map output records=113036  
  Input split bytes=210  
  Spilled Records=0  
  Failed Shuffles=0  
  Merged Map outputs=0  
  GC time elapsed (ms)=394  
  CPU time spent (ms)=12720  
  Physical memory (bytes) snapshot=483278848  
  Virtual memory (bytes) snapshot=3136581632  
  Total committed heap usage (bytes)=498073600  
File Input Format Counters  
  Bytes Read=0  
File Output Format Counters  
  Bytes Written=15339665  
23/07/27 20:23:34 INFO mapreduce.ImportJobBase: Transferred 14.629 MB in 82.1245 seconds (182.4076 KB/sec)  
23/07/27 20:23:34 INFO mapreduce.ImportJobBase: Retrieved 113036 records.  
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost:3306/project 1024 --table sales_csv --username root --  
password cloudera --target-dir /user/cloudera/my_project/sales_data.csv --split-by ID --m 2;  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
23/07/27 20:38:30 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0  
23/07/27 20:38:30 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.  
23/07/27 20:38:30 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.  
23/07/27 20:38:30 INFO tool.CodeGenTool: Beginning code generation  
23/07/27 20:38:31 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `sales_csv` AS t LIMIT 1  
23/07/27 20:38:31 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `sales_csv` AS t LIMIT 1  
23/07/27 20:38:31 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce  
Note: /tmp/sqoop-cloudera/compile/aaa67b2e2851168731d7e81625a5932b/sales_csv.java uses or overrides a deprecated API.  
Note: Recompile with -Xlint:deprecation for details.  
23/07/27 20:38:36 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/aaa67b2e2851168731d7e81625a5932b/  
sales_csv.jar  
23/07/27 20:38:37 WARN manager.MySQLManager: It looks like you are connecting from mysql.
```

Figure 6 shows the use of Sqoop to import data into HDFS.

3.2. DATA INGESTION INTO HIVE:

From HDFS we ingested the data into Hive for further analysis. To achieve this, we used the load data inpath to ingest into the Hive. For sure, a table of similar schema was created before the ingestion of data from HDFS to Hive. Figure 7 shows the syntax used for ingestion.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
hive> show tables
> ;
OK
tab name
Time taken: 0.598 seconds
hive> create table sales_csv (ID int, Date string, months varchar(11), Year string, cust_ID int, cust_Age int, age_group string, cust_gender string, Country string, state string, prod_cat string, sub_cat string, prod string, frame_size int, order_qty int, unit_cost int, unit_price int, cost int, profit int, revenue int)
> row format delimited
> fields terminated by ','
> TBLPROPERTIES("skip.header.line.count"="1");
OK
Time taken: 0.404 seconds
hive> LOAD DATA INPATH '/home/cloudera/my_project/sales_data.csv' into table sales_csv;
FAILED: SemanticException Line 1:17 Invalid path ''/home/cloudera/my_project/sales_data.csv'': No files matching path hdfs://quickstart.cloudera:8020/home/cloudera/my_project/sales_data.csv
hive> LOAD DATA INPATH '/user/cloudera/my_project/sales_data.csv' into table sales_csv;
Loading data to table project_1024.sales_csv
Table project_1024.sales_csv stats: [numFiles=2, totalSize=15339665]
OK
Time taken: 1.521 seconds
hive> select * from sales_csv limit 5;
OK
sales_csv.id    sales_csv.date    sales_csv.months    sales_csv.year    sales_csv.cust_id    sales_csv.cust_age    sales_csv.cust_gender    sales_csv.country    sales_csv.state    sales_csv.prod_cat    sales_csv.sub_cat    sales_csv.prod    sales_csv.frame_size    sales_csv.order_qty    sales_csv.unit_cost    sales_csv.unit_price    sales_csv.cost    sales_csv.profit    sales_csv.revenue
2    6/23/2014    June    2014    20364    61    senior    F    United States    California    Clothing    Short
3    5/20/2014    May    2014    20527    48    adult    F    United States    California    Bikes    Road Bikes    R
oad-550-W Yellow    NULL    42    1    713    1120    713    523
4    5/20/2016    May    2016    20527    48    adult    F    United States    California    Bikes    Road Bikes    R
oad-550-W Yellow    NULL    42    1    713    1120    713    261
5    6/26/2014    June    2014    12036    56    adult    F    Australia    Queensland    Accessories    Tires
and Tubes    Patch Kit/8 Patches    0    4    1    2    4    315    7
6    6/2/2014    June    2014    11211    48    adult    F    Canada    British Columbia    Accessories    Tires
and Tubes    Road Tire Tube    0    2    1    4    2    77    8
Time taken: 0.174 seconds Fetched: 5 row(s)
Browsing HDFS - Mozilla Firefox

```

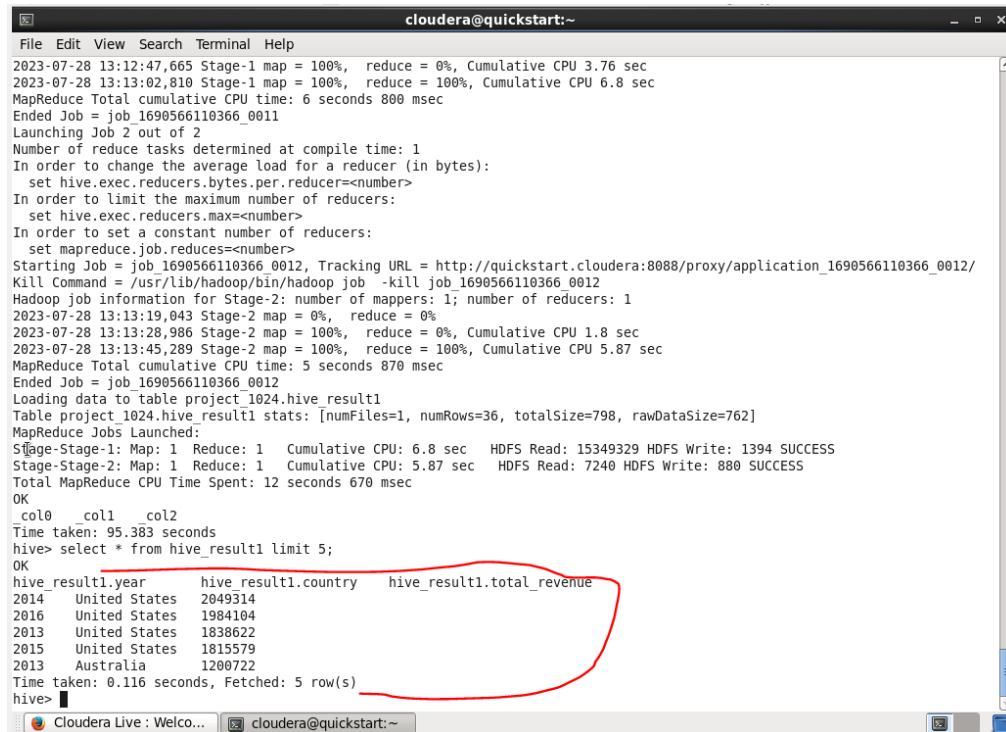
Figure 7 shows loading of data from HDFS into Hive

Currently we have successfully ingested our data into Hive for analysis. Here we will derive a few insights from our data. The purpose is to understand our data better. Furthermore, after generating results from Hive queries, we must store them and export the results into MySQL. The importance of this is to understand how to ingest data between HDFS frameworks.

So, to export the query results from hive to MySQL, then we must create individual result tables with the schema matching. Once the result is viewed, we create a table for it and store the result inside the table. The result table is different from the entire data table as shown in figure 9. To store the result differently in Hive, we used the syntax in figure 10.

4. DATA ANALYSIS USING HIVE:

After demonstrating how the data was transferred between Hive and MySQL, we will explain the insights derived from the queries.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
2023-07-28 13:12:47,665 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.76 sec  
2023-07-28 13:13:02,810 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.8 sec  
MapReduce Total cumulative CPU time: 6 seconds 800 msec  
Ended Job = job_1690566110366_0011  
Launching Job 2 out of 2  
Number of reduce tasks determined at compile time: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0012, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0012/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0012  
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1  
2023-07-28 13:13:19,043 Stage-2 map = 0%, reduce = 0%  
2023-07-28 13:13:28,986 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 1.8 sec  
2023-07-28 13:13:45,289 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 5.87 sec  
MapReduce Total cumulative CPU time: 5 seconds 870 msec  
Ended Job = job_1690566110366_0012  
Loading data to table project_1024.hive_result1  
Table project_1024.hive_result1 stats: [numFiles=1, numRows=36, totalSize=798, rawDataSize=762]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.8 sec HDFS Read: 15349329 HDFS Write: 1394 SUCCESS  
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 5.87 sec HDFS Read: 7240 HDFS Write: 880 SUCCESS  
Total MapReduce CPU Time Spent: 12 seconds 670 msec  
OK  
_col0 _col1 _col2  
Time taken: 95.383 seconds  
hive> select * from hive_result1 limit 5;  
OK  
hive_result1.year  hive_result1.country  hive_result1.total_revenue  
2014 United States  2049314  
2016 United States  1984104  
2013 United States  1838622  
2015 United States  1815579  
2013 Australia     1200722  
Time taken: 0.116 seconds, Fetched: 5 row(s)  
hive>
```

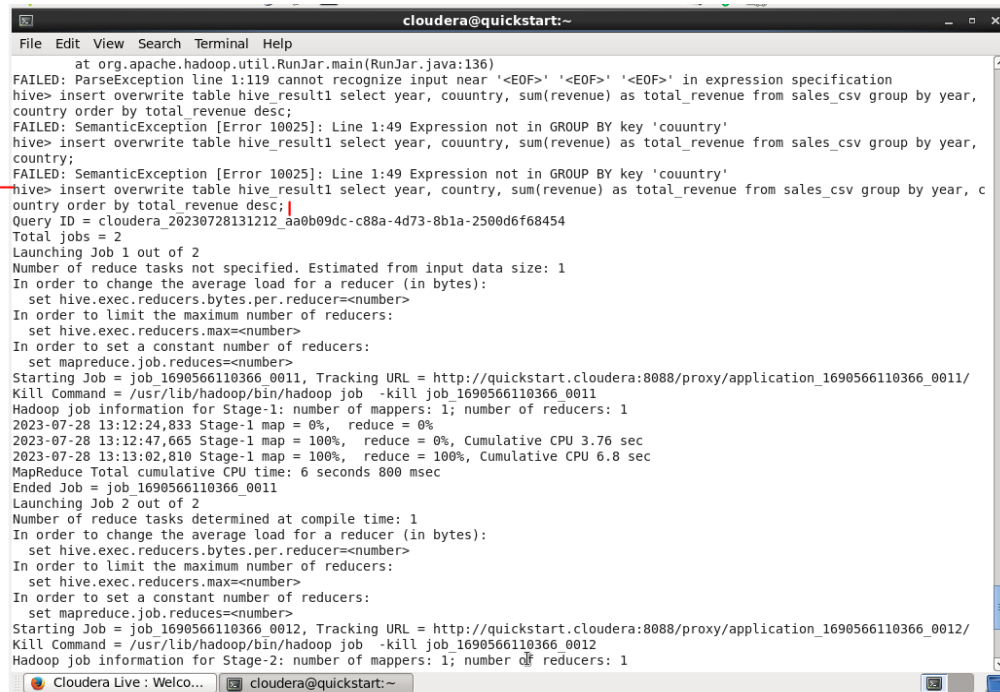
Figure 8 shows the query result.

```
Time taken: 59.176 seconds, Fetched: 36 row(s)  
hive> create table hive_result1 (year int, country string, total_revenue int);  
OK  
Time taken: 6.740 seconds
```

Figure 9 shows the creation of query result table for storing the result in Hive.

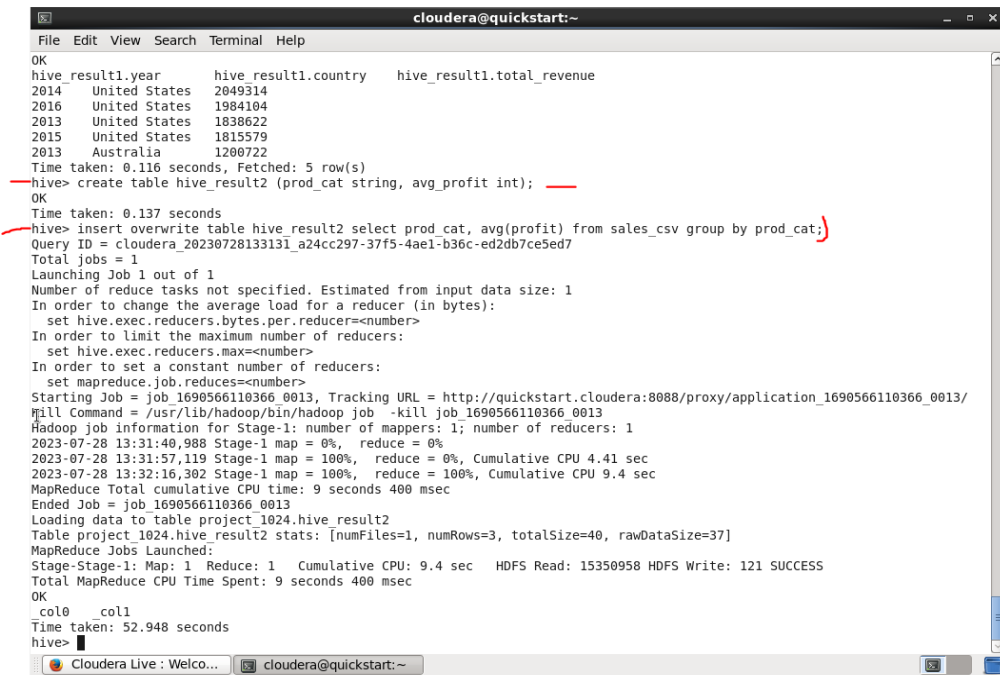
4.1. EXPORTING HIVE QUERY RESULTS TO HIVE RESULT TABLES:

We have created 5 different tables using hive query and inserting the data into those tables.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
at org.apache.hadoop.util.RunJar.main(RunJar.java:136)  
FAILED: ParseException line 1:119 cannot recognize input near '<EOF>' '<EOF>' '<EOF>' in expression specification  
hive> insert overwrite table hive_result1 select year, country, sum(revenue) as total_revenue from sales_csv group by year,  
country order by total_revenue desc;  
FAILED: SemanticException [Error 10025]: Line 1:49 Expression not in GROUP BY key 'country'  
hive> insert overwrite table hive_result1 select year, country, sum(revenue) as total_revenue from sales_csv group by year,  
country;  
FAILED: SemanticException [Error 10025]: Line 1:49 Expression not in GROUP BY key 'country'  
hive> insert overwrite table hive_result1 select year, country, sum(revenue) as total_revenue from sales_csv group by year, c  
ountry order by total_revenue desc;  
Query ID = cloudera_20230728131212_aa0b09dc-c88a-4d73-8b1a-2500d6f68454  
Total jobs = 2  
Launching Job 1 out of 2  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0011, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0011/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0011  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 13:12:24,833 Stage-1 map = 0%, reduce = 0%  
2023-07-28 13:12:47,665 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.76 sec  
2023-07-28 13:13:02,810 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.8 sec  
MapReduce Total cumulative CPU time: 6 seconds 800 msec  
Ended Job = job_1690566110366_0011  
Launching Job 2 out of 2  
Number of reduce tasks determined at compile time: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0012, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0012/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0012  
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
```

Figure 10 show syntax for inserting query result into a separate table



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
OK  
hive_result1.year      hive_result1.country  hive_result1.total_revenue  
2014    United States  2049314  
2016    United States  1984104  
2013    United States  1838622  
2015    United States  1815579  
2013    Australia      1200722  
Time taken: 0.116 seconds, Fetched: 5 row(s)  
hive> create table hive_result2 (prod_cat string, avg_profit int);  
OK  
Time taken: 0.137 seconds  
hive> insert overwrite table hive_result2 select prod_cat, avg(profit) from sales_csv group by prod_cat;  
Query ID = cloudera_20230728133131_a24cc297-37f5-4ae1-b36c-ed2db7ce5ed7  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0013, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0013/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0013  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 13:31:40,988 Stage-1 map = 0%, reduce = 0%  
2023-07-28 13:31:57,119 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.41 sec  
2023-07-28 13:32:16,302 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.4 sec  
MapReduce Total cumulative CPU time: 9 seconds 400 msec  
Ended Job = job_1690566110366_0013  
Loading data to table project_1024.hive_result2  
Table project_1024.hive_result2 stats: [numFiles=1, numRows=3, totalSize=40, rawDataSize=37]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.4 sec HDFS Read: 15350958 HDFS Write: 121 SUCCESS  
Total MapReduce CPU Time Spent: 9 seconds 400 msec  
OK  
col0    col1  
Time taken: 52.948 seconds  
hive>
```

Figure 11 shows creation of hive_result2 and insert query result into it.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Time taken: 0.116 seconds, Fetched: 5 row(s)  
hive> create table hive_result2 (prod_cat string, avg_profit int);  
OK  
Time taken: 0.137 seconds  
hive> insert overwrite table hive_result2 select prod_cat, avg(profit) from sales_csv group by prod_cat;  
Query ID = cloudera_20230728133131_a24cc297-37f5-4ae1-b36c-ed2db7ce5ed7  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0013, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0013/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0013  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 13:31:40,988 Stage-1 map = 0%, reduce = 0%  
2023-07-28 13:31:57,119 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.41 sec  
2023-07-28 13:32:16,302 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.4 sec  
MapReduce Total cumulative CPU time: 9 seconds 400 msec  
Ended Job = job_1690566110366_0013  
Loading data to table project_1024.hive_result2  
Table project_1024.hive_result2 stats: [numFiles=1, numRows=3, totalSize=40, rawDataSize=37]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.4 sec HDFS Read: 15350958 HDFS Write: 121 SUCCESS  
Total MapReduce CPU Time Spent: 9 seconds 400 msec  
OK  
_col0 _col1  
Time taken: 52.948 seconds  
hive> select * from hive_result2;  
OK  
hive_result2.prod_cat hive_result2.avg_profit  
Accessories 288  
Bikes 1588  
Clothing 340  
Time taken: 0.089 seconds, Fetched: 3 row(s)  
hive>
```

Figure 12 shows result insertion into hive_result2 and result is displayed.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
hive> select * from hive_result2;  
OK  
hive_result2.prod_cat hive_result2.avg_profit  
Accessories 288  
Bikes 1588  
Clothing 340  
Time taken: 0.089 seconds, Fetched: 3 row(s)  
hive> create table hive_result3 (country string, state string, no_sales int);  
OK  
Time taken: 0.148 seconds  
hive> insert overwrite table hive_result3 select country, state, count(*) as no_sales from sales_csv group by country, state;  
Query ID = cloudera_20230728134141_2f0dccc34-0f06-4555-81e4-68ac911a3da0  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0014, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0014/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0014  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 13:41:28,685 Stage-1 map = 0%, reduce = 0%  
2023-07-28 13:41:43,900 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.98 sec  
2023-07-28 13:42:01,344 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 7.78 sec  
MapReduce Total cumulative CPU time: 7 seconds 780 msec  
Ended Job = job_1690566110366_0014  
Loading data to Table project_1024.hive_result3  
Table project_1024.hive_result3 stats: [numFiles=1, numRows=53, totalSize=1313, rawDataSize=1260]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.78 sec HDFS Read: 15350869 HDFS Write: 1396 SUCCESS  
Total MapReduce CPU Time Spent: 7 seconds 780 msec  
OK  
_col0 _col1 _col2  
Time taken: 48.342 seconds  
hive>
```

Figure 13 shows creation hive_result3 and insertion of result query syntax

```

cloudera@quickstart:~
File Edit View Search Terminal Help
OK
hive_result3.country    hive_result3.state    hive_result3.no_sales
Australia    New South Wales    10412
Australia    Queensland    5220
Australia    South Australia    1564
Australia    Tasmania    724
Australia    Victoria    6016
Canada    Alberta    56
Canada    British Columbia    14116
Canada    Ontario    6
France    Charente-Maritime    148
France    Essonne    994
France    Garonne (Haute)    208
France    Hauts de Seine    1084
France    Loir et Cher    120
France    Loiret    382
France    Moselle    386
France    Nord    1670
France    Pas de Calais    90
France    Seine (Paris)    2328
France    Seine Saint Denis    1684
France    Seine et Marne    394
France    Somme    134
France    Val d'Oise    264
France    Val de Marne    158
France    Yveline    954
Germany    Bayern    1426
Germany    Brandenburg    198
Germany    Hamburg    1836
Germany    Hessen    2384
Germany    Nordrhein-Westfalen    2484
Germany    Saarland    2770
United Kingdom    England    13620
United States    Alabama    4
United States    Arizona    4
United States    California    22450
United States    Florida    14
United States    Georgia    8
United States    Illinois    28

```

Figure 14 shows the display of hive_result3 table content.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:389)
at org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:781)
at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:699)
at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:634)
at sun.reflect.NativeMethodAccessorImpl.invoke(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
FAILED: ParseException line 1:60 cannot recognize input near 'from' 'sales_csv' 'order' in selection target
hive> insert overwrite table hive_result4 select prod, order_qty from sales_csv order by order_qty;
Query ID = cloudera_20230728135050_cb235c4a-6111-4b29-a473-69dad1ca0754
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1690566110366_0016, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0016/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0016
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2023-07-28 13:50:58,202 Stage-1 map = 0%, reduce = 0%
2023-07-28 13:51:11,575 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 5.85 sec
2023-07-28 13:51:28,982 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 11.47 sec
MapReduce Total cumulative CPU time: 11 seconds 470 msec
Ended Job = job_1690566110366_0016
Loading data to table project_1024.hive_result4
Table: project_1024.hive_result4 stats: [numFiles=1, numRows=113034, totalSize=2380577, rawDataSize=2267543]
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 11.47 sec HDFS Read: 15348637 HDFS Write: 2380667 SUCCESS
Total MapReduce CPU Time Spent: 11 seconds 470 msec
OK
prod order_qty
Time taken: 47.574 seconds
hive>

```

Figure 15 shows creation hive_result4 and insertion of result query syntax

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Time taken: 0.111 seconds, Fetched: 3 row(s)  
hive> create table hive_result5 (prod_cat string, sub_cat string, cost int, revenue int);  
OK  
Time taken: 0.157 seconds  
hive> insert overwrite table hive_result5 select prod_cat, sub_cat, sum(cost) as total_cost, sum(revenue) as total_revenue fr  
om sales_csv group by prod_cat, sub_cat;  
FAILED: SemanticException [Error 10004]: Line 1:141 Invalid table alias or column reference 'prod_cat': (possible column names  
are: id, date, months, year, cust_id, cust_age, age_group, cust_gender, country, state, prod_cat, sub_cat, prod, frame_size,  
order_qty, unit_cost, unit_price, cost, profit, revenue)  
hive> insert overwrite table hive_result5 select prod_cat, sub_cat, sum(cost) as total_cost, sum(revenue) as total_revenue fr  
om sales_csv group by prod_cat, sub_cat;  
Query ID = cloudera_20230728140202_ed7b3d8c-a628-4fc5-8288-6048980c3cba  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0018, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0018/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0018  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 14:03:01,206 Stage-1 map = 0%, reduce = 0%  
2023-07-28 14:03:12,068 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.97 sec  
2023-07-28 14:03:28,249 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 8.08 sec  
MapReduce Total cumulative CPU time: 8 seconds 80 msec  
Ended Job = job_1690566110366_0018  
Loading data to table project_1024.hive_result5  
Table project_1024.hive_result5 stats: [numFiles=1, numRows=17, totalSize=593, rawDataSize=576]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 8.08 sec HDFS Read: 15351340 HDFS Write: 675 SUCCESS  
Total MapReduce CPU Time Spent: 8 seconds 80 msec  
OK  
_col0 _col1 _col2 _col3  
Time taken: 40.619 seconds
```

Figure 16 shows creation hive_result5 and insertion of result query syntax.

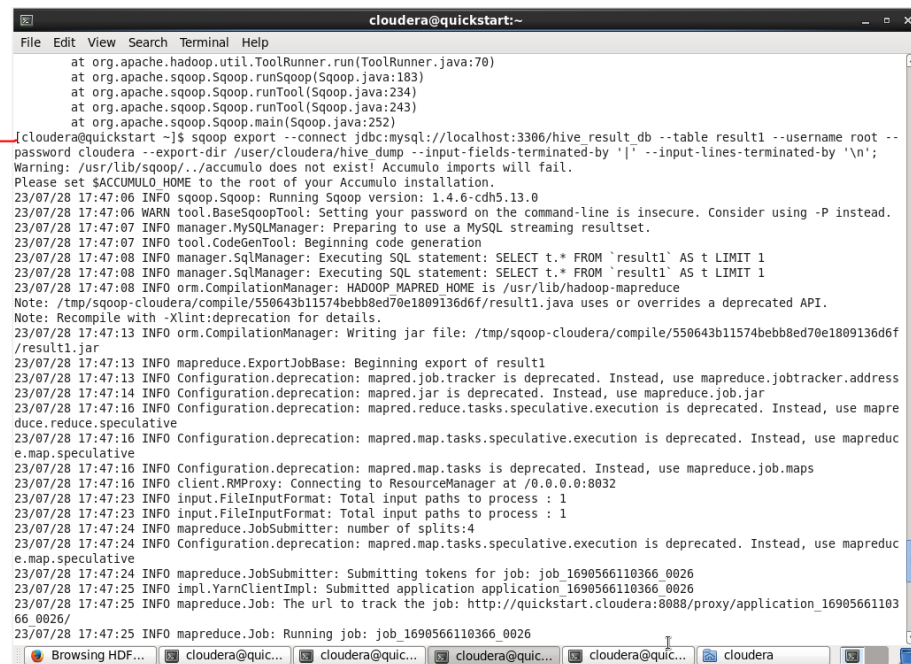
```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Query ID = cloudera_20230728140202_ed7b3d8c-a628-4fc5-8288-6048980c3cba  
Total jobs = 1  
Launching Job 1 out of 1  
Number of reduce tasks not specified. Estimated from input data size: 1  
In order to change the average load for a reducer (in bytes):  
  set hive.exec.reducers.bytes.per.reducer=<number>  
In order to limit the maximum number of reducers:  
  set hive.exec.reducers.max=<number>  
In order to set a constant number of reducers:  
  set mapreduce.job.reduces=<number>  
Starting Job = job_1690566110366_0018, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0018/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0018  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1  
2023-07-28 14:03:01,206 Stage-1 map = 0%, reduce = 0%  
2023-07-28 14:03:12,068 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.97 sec  
2023-07-28 14:03:28,249 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 8.08 sec  
MapReduce Total cumulative CPU time: 8 seconds 80 msec  
Ended Job = job_1690566110366_0018  
Loading data to table project_1024.hive_result5  
Table project_1024.hive_result5 stats: [numFiles=1, numRows=17, totalSize=593, rawDataSize=576]  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 8.08 sec HDFS Read: 15351340 HDFS Write: 675 SUCCESS  
Total MapReduce CPU Time Spent: 8 seconds 80 msec  
OK  
_col0 _col1 _col2 _col3  
Time taken: 40.619 seconds  
hive> select * from hive_result5 order by total revenue desc limit 5;  
FAILED: SemanticException [Error 10004]: Line 1:37 Invalid table alias or column reference 'total_revenue': (possible column  
names are: prod_cat, sub_cat, cost, revenue)  
hive> select * from hive_result5 limit 5;  
OK  
hive_result5.prod_cat  hive_result5.sub_cat  hive_result5.cost  hive_result5.revenue  
Accessories  Bike Racks  213345  517800  
Accessories  Bike Stands  142140  344075  
Accessories  Bottles and Cages  598576  1409174  
Accessories  Cleaners  82930  198821  
Accessories  Fenders 496819 1245733  
Time taken: 0.133 seconds, Fetched: 5 row(s)  
hive>
```

Figure 16 shows creation hive_result5 results.

Previously we have been saving our different Hive results by *inserting overwrite--* syntax as shown in the above figures because we are preparing them for export to MySQL. Remember we are dealing with structured data, therefore the table schemas between origin and destination must be the same. Here the data is cleaned further by removing the pipe symbol before saving it into HDFS.

In our project we created `hive_result_db` database and created respective result tables with exact matching hive schemas in the above-mentioned database. Since we are moving data between HDFS frameworks, this means we must save (dump) individual results into HDFS first, then from there we export to MySQL.

Note, before exporting to MySQL, we had to save into HDFS using the *insert overwrite directory to save* in HDFS. Figure 17 shows how we exported our Hive data (`hive_result1`) to MySQL (`result1`). The consequent figures will show the syntax for saving the Hive result into HDFS.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)  
at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:183)  
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:234)  
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:243)  
at org.apache.sqoop.Sqoop.main(Sqoop.java:252)  
cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/hive_result_db --table result1 --username root --  
password cloudera --export-dir /user/cloudera/hive_dump --input-fields-terminated-by '|' --input-lines-terminated-by '\n';  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
23/07/28 17:47:06 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0  
23/07/28 17:47:06 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.  
23/07/28 17:47:07 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.  
23/07/28 17:47:07 INFO tool.CodeGenTool: Beginning code generation  
23/07/28 17:47:08 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `result1` AS t LIMIT 1  
23/07/28 17:47:08 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `result1` AS t LIMIT 1  
23/07/28 17:47:08 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce  
Note: /tmp/sqoop-cloudera/compile/550643b11574bebb8ed70e1809136d6f/result1.java uses or overrides a deprecated API.  
Note: Recompile with -Xlint:deprecation for details.  
23/07/28 17:47:13 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/550643b11574bebb8ed70e1809136d6f/  
result1.jar  
23/07/28 17:47:13 INFO mapreduce.ExportJobBase: Beginning export of result1  
23/07/28 17:47:13 INFO Configuration.deprecation: mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address  
23/07/28 17:47:14 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar  
23/07/28 17:47:16 INFO Configuration.deprecation: mapred.reduce.tasks.speculative.execution is deprecated. Instead, use mapred  
duce.reduce.speculative  
23/07/28 17:47:16 INFO Configuration.deprecation: mapred.map.tasks.speculative.execution is deprecated. Instead, use mapred  
e.map.speculative  
23/07/28 17:47:16 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps  
23/07/28 17:47:16 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032  
23/07/28 17:47:23 INFO input.FileInputFormat: Total input paths to process : 1  
23/07/28 17:47:23 INFO input.FileInputFormat: Total input paths to process : 1  
23/07/28 17:47:24 INFO mapreduce.JobSubmitter: number of splits:4  
23/07/28 17:47:24 INFO Configuration.deprecation: mapred.map.tasks.speculative.execution is deprecated. Instead, use mapred  
e.map.speculative  
23/07/28 17:47:24 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1690566110366_0026  
23/07/28 17:47:25 INFO impl.YarnClientImpl: Submitted application application_1690566110366_0026  
23/07/28 17:47:25 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_16905661103  
66_0026/  
23/07/28 17:47:25 INFO mapreduce.Job: Running job: job_1690566110366_0026
```

Figure 17 shows export syntax to MySQL specified DB and table.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
ERROR 1146 (42S02): Table 'hive_result_db.results' doesn't exist
mysql> select * from result1;
Empty set (0.00 sec)

mysql> select * from result1;
+-----+-----+-----+
| year | country | total_revenue |
+-----+-----+-----+
| 2013 | Canada | 610331 |
| 2014 | France | 569858 |
| 2016 | France | 535775 |
| 2013 | Germany | 497589 |
| 2015 | Germany | 478500 |
| 2013 | France | 455517 |
| 2015 | France | 454301 |
| 2012 | United States | 257037 |
| 2012 | Australia | 247307 |
| 2011 | Australia | 245987 |
| 2011 | United States | 237599 |
| 2012 | United Kingdom | 84024 |
| 2011 | United Kingdom | 89421 |
| 2011 | Canada | 67586 |
| 2011 | France | 65788 |
| 2012 | France | 61310 |
| 2012 | Canada | 57794 |
| 2012 | Germany | 56977 |
| 2011 | Germany | 56559 |
| 2016 | Canada | 752441 |
| 2014 | United States | 2049314 |
| 2014 | Canada | 739327 |
| 2016 | United Kingdom | 735113 |
| 2014 | United Kingdom | 732016 |
| 2014 | Germany | 648943 |
| 2016 | Germany | 639400 |
| 2013 | United Kingdom | 638193 |
| 2015 | Canada | 628493 |
| 2015 | United Kingdom | 623588 |
| 2016 | United States | 1984104 |
| 2013 | United States | 1830622 |

```

Figure 18 shows the result exported Into MySQL result1 table.

4.2. EXPORTING DATA FROM HIVE TO MySQL:

Like we mentioned before, the insert syntax for saving Hive results into HDFS is clearly shown in figure 19. While selecting all the results in Hive_result table, it also removes the Pipe '|' delimiter then save the data in HDFS. This makes it easier for each table to be exported into MySQL.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1690566110366_0025, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0025/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0025
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2023-07-28 17:44:34,446 Stage-1 map = 0%, reduce = 0%
2023-07-28 17:44:44,341 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.5 sec
MapReduce Total cumulative CPU time: 2 seconds 500 msec
Ended Job = job_1690566110366_0025
Stage-3 is selected by condition resolver.
Stage-2 is filtered out by condition resolver.
Stage-4 is filtered out by condition resolver.
Moving data to: hdfs://quickstart.cloudera:8020/user/cloudera/hive_dump/.hive-staging_hive_2023-07-28_17-44-19_378_7849035855
584837620-1/-ext-10000
Moving data to: /user/cloudera/hive_dump
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Cumulative CPU: 2.5 sec HDFS Read: 4332 HDFS Write: 798 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 500 msec
OK
hive> select * from hive_result1;
hive_result1.year    hive_result1.country  hive_result1.total_revenue
Time taken: 26.145 seconds
hive> select * from hive_result2;
OK
hive_result2.prod_cat  hive_result2.avg_profit
Accessories           288
Bikes                 1588
Clothing              340
Time taken: 0.149 seconds, Fetched: 3 row(s)
hive> insert overwrite directory '/user/cloudera/hive_dump' row format delimited fields terminated by '|' select * from hive_result2;
Query ID = cloudera_20230728180000_7c13d3df-d622-40ef-b1a6-82d56afdaeef
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1690566110366_0027, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0027/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0027
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2023-07-28 18:00:17,824 Stage-1 map = 0%, reduce = 0%
2023-07-28 18:00:29,037 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.03 sec

```

Figure 19 shows syntax for saving Hive_result2 data into HDFS.

For table result2 as an example, we could see that in figure 19, we saved the hive data in HDFS, in figure 20, we exported hive result into respective MySQL table and made sure that the appropriate delimiters are specified.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
File System Counters
  FILE: Number of bytes read=0
  FILE: Number of bytes written=683712
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=692
  HDFS: Number of bytes written=0
  HDFS: Number of read operations=16
  HDFS: Number of large read operations=0
  HDFS: Number of write operations=0
Job Counters
  Launched map tasks=4
  Data-local map tasks=4
  Total time spent by all maps in occupied slots (ms)=117509
  Total time spent by all reduces in occupied slots (ms)=0
  Total time spent by all map tasks (ms)=117509
  Total vcore-milliseconds taken by all map tasks=117509
  Total megabyte-milliseconds taken by all map tasks=120329216
Map-Reduce Framework
  Map input records=3
  Map output records=3
  Input split bytes=580
  Spilled Records=0
  Failed Shuffles=0
  Merged Map outputs=0
  GC time elapsed (ms)=478
  CPU time spent (ms)=6990
  Physical memory (bytes) snapshot=660807680
  Virtual memory (bytes) snapshot=6274813952
  Total committed heap usage (bytes)=499122176
File Input Format Counters
  Bytes Read=0
File Output Format Counters
  Bytes Written=0
23/07/28 18:04:35 INFO mapreduce.ExportJobBase: Transferred 692 bytes in 54.1608 seconds (12.7768 bytes/sec)
23/07/28 18:04:35 INFO mapreduce.ExportJobBase: Exported 3 records.
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/hive_result_db --table result2 --username root --
password cloudera --export-dir /user/cloudera/hive_dump --input-fields-terminated-by '|' --input-lines-terminated-by '\n';

```

Figure 20 shows the syntax for exporting data into MySQL.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
+-----+-----+-----+
| 2012 | United Kingdom | 84024 |
| 2011 | United Kingdom | 80421 |
| 2011 | Canada | 67586 |
| 2011 | France | 65788 |
| 2012 | France | 61310 |
| 2012 | Canada | 57794 |
| 2012 | Germany | 56977 |
| 2011 | Germany | 56559 |
| 2016 | Canada | 752441 |
| 2014 | United States | 2049314 |
| 2014 | Canada | 739327 |
| 2016 | United Kingdom | 735113 |
| 2014 | United Kingdom | 732816 |
| 2014 | Germany | 648943 |
| 2016 | Germany | 639400 |
| 2013 | United Kingdom | 638193 |
| 2015 | Canada | 628493 |
| 2015 | United Kingdom | 623588 |
| 2016 | United States | 1904104 |
| 2013 | United States | 1838622 |
| 2015 | United States | 1815579 |
| 2013 | Australia | 1200722 |
| 2015 | Australia | 1176899 |
| 2016 | Australia | 1132348 |
| 2014 | Australia | 1120450 |
+-----+-----+-----+
36 rows in set (0.00 sec)

mysql> select * from result2;
+-----+-----+
| prod_cat | avg_profit |
+-----+-----+
| Accessories | 288 |
| Bikes | 1588 |
| Clothing | 340 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

Figure 21 shows the result of MySQL data which matches Hive_result.


```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
United States Missouri 6  
United States Montana 6  
United States New York 20  
United States North Carolina 4  
United States Ohio 28  
United States Oregon 5286  
United States South Carolina 10  
United States Texas 30  
United States Utah 10  
United States Virginia 4  
United States Washington 11262  
United States Wyoming 8  
Time taken: 0.142 seconds, Fetched: 53 row(s)  
hive> insert overwrite directory '/user/cloudera/hive_dump' row format delimited fields terminated by '|' select * from hive_  
result3;  
Query ID = cloudera_20230728181515_fb194bb3-a491-4b80-a5f7-b2f9fb83195c  
Total jobs = 3  
Launching Job 1 out of 3  
Number of reduce tasks is set to 0 since there's no reduce operator  
Starting Job = job 1690566110366_0029, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0029/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job 1690566110366_0029  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0  
2023-07-28 18:15:25,590 Stage-1 map = 0%, reduce = 0%  
2023-07-28 18:15:36,590 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.64 sec  
MapReduce Total cumulative CPU time: 2 seconds 640 msec  
Ended Job = job 1690566110366_0029  
Stage-3 is selected by condition resolver.  
Stage-2 is filtered out by condition resolver.  
Stage-4 is filtered out by condition resolver.  
Moving data to: hdfs://quickstart.cloudera:8020/user/cloudera/hive_dump/.hive-staging_hive_2023-07-28_18-15-07_169_1224548807  
998368762-1/-ext-10000  
Moving data to: /user/cloudera/hive_dump  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Cumulative CPU: 2.64 sec HDFS Read: 4756 HDFS Write: 1313 SUCCESS  
Total MapReduce CPU Time Spent: 2 seconds 640 msec  
OK  
hive_result3.country hive_result3.state hive_result3.no_sales  
Time taken: 31.73 seconds  
hive>
```

Figure 22 shows syntax for saving hive_result3 data into HDFS.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
File System Counters  
FILE: Number of bytes read=0  
FILE: Number of bytes written=683732  
FILE: Number of read operations=0  
FILE: Number of large read operations=0  
FILE: Number of write operations=0  
HDFS: Number of bytes read=4125  
HDFS: Number of bytes written=0  
HDFS: Number of read operations=19  
HDFS: Number of large read operations=0  
HDFS: Number of write operations=0  
Job Counters  
Launched map tasks=4  
Data-local map tasks=4  
Total time spent by all maps in occupied slots (ms)=114559  
Total time spent by all reduces in occupied slots (ms)=0  
Total time spent by all map tasks (ms)=114559  
Total vcore-milliseconds taken by all map tasks=114559  
Total megabyte-milliseconds taken by all map tasks=117308416  
Map-Reduce Framework  
Map input records=53  
Map output records=53  
Input split bytes=661  
Spilled Records=0  
Failed Shuffles=0  
Merged Map outputs=0  
GC time elapsed (ms)=721  
CPU time spent (ms)=6240  
Physical memory (bytes) snapshot=672002048  
Virtual memory (bytes) snapshot=6267613184  
Total committed heap usage (bytes)=566231040  
File Input Format Counters  
Bytes Read=0  
File Output Format Counters  
Bytes Written=0  
23/07/28 18:18:28 INFO mapreduce.ExportJobBase: Transferred 4.0283 KB in 53.2484 seconds (77.4672 bytes/sec)  
23/07/28 18:18:28 INFO mapreduce.ExportJobBase: Exported 53 records.  
cloudera@quickstart ~$ sqoop export --connect jdbc:mysql://localhost:3306/hive result db --table result3 --username root --  
password cloudera --export-dir /user/cloudera/hive_dump --input-fields-terminated-by '|' --input-lines-terminated-by '\n';
```

Figure 23 syntax shows successful export of data into MySQL result3 table.

```

cloudera@quickstart:~
File Edit View Search Terminal Help
Query OK, 0 rows affected (0.05 sec)

mysql> select * from result3;
+-----+-----+-----+
| country | state | no_sales |
+-----+-----+-----+
| United States | Missouri | 6 |
| United States | Montana | 6 |
| United States | New York | 20 |
| United States | North Carolina | 4 |
| United States | Ohio | 28 |
| United States | Oregon | 5286 |
| United States | South Carolina | 10 |
| United States | Texas | 30 |
| United States | Utah | 10 |
| United States | Virginia | 4 |
| United States | Washington | 11262 |
| United States | Wyoming | 8 |
| Australia | New South Wales | 19412 |
| Australia | Queensland | 5220 |
| Australia | South Australia | 1564 |
| Australia | Tasmania | 724 |
| Australia | Victoria | 6016 |
| Canada | Alberta | 56 |
| Canada | British Columbia | 14116 |
| Canada | Ontario | 6 |
| France | Charente-Maritime | 148 |
| France | Essonne | 994 |
| France | Garonne (Haute) | 208 |
| France | Hauts de Seine | 1084 |
| France | Loir et Cher | 120 |
| France | Loiret | 382 |
| Germany | Saarland | 2770 |
| United Kingdom | England | 13620 |
| United States | Alabama | 4 |
| United States | Arizona | 4 |
| United States | California | 22450 |
| United States | Florida | 14 |
| United States | Georgia | 8 |

```

Figure 24 shows result of MySQL result3 table successful transferred data

```

cloudera@quickstart:~
File Edit View Search Terminal Help
Women's Mountain Shorts 62
Road-350-W Yellow 62
Road-350-W Yellow 62
Short-Sleeve Classic Jersey 62
Touring-3000 Blue 62
Women's Mountain Shorts 62
Long-Sleeve Logo Jersey 62
Classic Vest 62
Mountain-200 Silver 62
Classic Vest 62
Road-150 Red 62
Racing Socks 62
Road-250 Black 62
Road-150 Red 62
Road-350-W Yellow 62
Road-550-W Yellow 62
Touring-1000 Yellow 62
Road-150 Red 62
Touring-2000 Blue 62
Touring-3000 Blue 62
Time taken: 0.347 seconds, Fetched: 113034 row(s)
hive> select * from hive_result4 limit 5;
OK
hive_result4.prod      hive_result4.order_qty
Long-Sleeve Logo Jersey 0
Touring-2000 Blue      0
Touring-2000 Blue      0
Short-Sleeve Classic Jersey 0
Road-350-W Yellow      0
Time taken: 0.171 seconds, Fetched: 5 row(s)
hive> insert overwrite directory '/user/cloudera/hive_dump' row format delimited fields terminated by '|' select * from hive_result4;
Query ID = cloudera_20230728182323_65fa7fc9-8014-4618-9232-955f32434bc3
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1690566110366_0031, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0031/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0031
cloudera@quickstart:~

```

Figure 25 shows the syntax for saving Hive_result4 into HDFS.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
File System Counters  
  FILE: Number of bytes read=0  
  FILE: Number of bytes written=683692  
  FILE: Number of read operations=0  
  FILE: Number of large read operations=0  
  FILE: Number of write operations=0  
  HDFS: Number of bytes read=2397637  
  HDFS: Number of bytes written=0  
  HDFS: Number of read operations=19  
  HDFS: Number of large read operations=0  
  HDFS: Number of write operations=0  
Job Counters  
  Launched map tasks=4  
  Data-local map tasks=4  
  Total time spent by all maps in occupied slots (ms)=130423  
  Total time spent by all reduces in occupied slots (ms)=0  
  Total time spent by all map tasks (ms)=130423  
  Total vcore-milliseconds taken by all map tasks=130423  
  Total megabyte-milliseconds taken by all map tasks=133553152  
Map-Reduce Framework  
  Map input records=113034  
  Map output records=113034  
  Input split bytes=661  
  Spilled Records=0  
  Failed Shuffles=0  
  Merged Map outputs=0  
  GC time elapsed (ms)=682  
  CPU time spent (ms)=18510  
  Physical memory (bytes) snapshot=756244480  
  Virtual memory (bytes) snapshot=6278340608  
  Total committed heap usage (bytes)=563085312  
File Input Format Counters  
  Bytes Read=0  
File Output Format Counters  
  Bytes Written=0  
23/07/28 18:50:43 INFO mapreduce.ExportJobBase: Transferred 2.2866 MB in 59.0824 seconds (39.6301 KB/sec)  
23/07/28 18:50:43 INFO mapreduce.ExportJobBase: Exported 113034 records.  
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/hive_result_db --table result4 --username root --  
password cloudera --export-dir /user/cloudera/hive_dump --input-fields-terminated-by '|' --input-lines-terminated-by '\n';
```

Figure 26 shows successful export of hive_result4 from HDFS to MySQL result4 table

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Mountain Tire Tube | 9 |  
Water Bottle - 30 oz. | 9 |  
Water Bottle - 30 oz. | 9 |  
AWC Logo Cap | 9 |  
Touring Tire Tube | 9 |  
Water Bottle - 30 oz. | 9 |  
Mountain Bottle Cage | 9 |  
Fender Set - Mountain | 9 |  
Fender Set - Mountain | 9 |  
Road Bottle Cage | 9 |  
Touring Tire | 9 |  
Fender Set - Mountain | 9 |  
Touring Tire Tube | 9 |  
Bike Wash - Dissolver | 9 |  
Mountain Tire Tube | 9 |  
Mountain Tire Tube | 9 |  
Bike Wash - Dissolver | 9 |  
Water Bottle - 30 oz. | 9 |  
AWC Logo Cap | 9 |  
Fender Set - Mountain | 9 |  
Fender Set - Mountain | 9 |  
Fender Set - Mountain | 9 |  
Fender Set - Mountain | 9 |  
-----+-----+-----  
113034 rows in set (0.19 sec)  
  
mysql> select * from result4 limit 5;  
+-----+-----+  
| prod | order_qty |  
+-----+-----+  
| Road Tire Tube | 22 |  
| Bike Wash - Dissolver | 22 |  
| Water Bottle - 30 oz. | 22 |  
| Touring Tire | 22 |  
| Patch Kit/8 Patches | 22 |  
+-----+-----+  
5 rows in set (0.00 sec)  
  
mysql>
```

Figure 27 shows successful display of exported hive table into MySQL.

The screenshot shows a terminal window titled 'cloudera@quickstart:~'. It displays the output of a Hive query that exports data from the 'hive_result5' table to HDFS. The output includes a list of items and their counts, followed by Hive execution details such as 'Time taken: 0.267 seconds, Fetched: 17 row(s)', 'Query ID = cloudera_20230728183838_2b38585f-5a69-4dc5-9b4a-98c7abd9aa58', and 'Total jobs = 3'. It also shows the progress of the MapReduce job, including 'MapReduce Total cumulative CPU time: 2 seconds 450 msec' and 'Ended Job = job_1690566110366_0035'. The final output is a list of items and their counts, followed by 'Time taken: 30.058 seconds'. The terminal window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The taskbar at the bottom shows several open windows, including 'Browsing HDF...' and 'cloudera@quic...'.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
Accessories Helmets 425530 2241977  
Accessories Hydration Packs 417067 988750  
Accessories Tires and Tubes 1943112 4670744  
Bikes Mountain Bikes 17477308 2105047  
Bikes Road Bikes 25502270 3948528  
Bikes Touring Bikes 6415790 2216878  
Clothing Caps 470856 548777  
Clothing Gloves 64664 510744  
Clothing Jerseys 311652 1157255  
Clothing Shorts 125510 587195  
Clothing Socks 10342 356459  
Clothing Vests 66336 528655  
Time taken: 0.267 seconds, Fetched: 17 row(s)  
hive> insert overwrite directory '/user/cloudera/hive_dump' row format delimited fields terminated by '|' select * from hive_  
result5;  
Query ID = cloudera_20230728183838_2b38585f-5a69-4dc5-9b4a-98c7abd9aa58  
Total jobs = 3  
Launching Job 1 out of 3  
Number of reduce tasks is set to 0 since there's no reduce operator  
Starting Job = job_1690566110366_0035, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1690566110366_0035/  
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1690566110366_0035  
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0  
2023-07-28 18:39:09,402 Stage-1 map = 0%, reduce = 0%  
2023-07-28 18:39:19,253 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.45 sec  
MapReduce Total cumulative CPU time: 2 seconds 450 msec  
Ended Job = job_1690566110366_0035  
Stage-3 is selected by condition resolver.  
Stage-2 is filtered out by condition resolver.  
Stage-4 is filtered out by condition resolver.  
Moving data to: hdfs://quickstart.cloudera:8020/user/cloudera/hive_dump/.hive-staging_hive_2023-07-28_18-38-50_587_7430097742  
782524128-1/-ext-10000  
Moving data to: /user/cloudera/hive_dump  
MapReduce Jobs Launched:  
Stage-Stage-1: Map: 1 Cumulative CPU: 2.45 sec HDFS Read: 4151 HDFS Write: 593 SUCCESS  
Total MapReduce CPU Time Spent: 2 seconds 450 msec  
OK  
hive_result5.prod_cat hive_result5.sub_cat hive_result5.cost hive_result5.revenue  
Time taken: 30.058 seconds  
Browsing HDFS - Mozilla Firefox
```

Figure 28 shows syntax for saving hive_result5 data into HDFS

The screenshot shows a terminal window titled 'cloudera@quickstart:~'. It displays the output of a Hive query that exports data from the 'hive_result5' table to MySQL. The output includes a list of items and their counts, followed by Hive execution details such as 'Time taken: 0.267 seconds, Fetched: 17 row(s)', 'Query ID = cloudera_20230728183838_2b38585f-5a69-4dc5-9b4a-98c7abd9aa58', and 'Total jobs = 3'. It also shows the progress of the MapReduce job, including 'MapReduce Total cumulative CPU time: 2 seconds 450 msec' and 'Ended Job = job_1690566110366_0035'. The final output is a list of items and their counts, followed by 'Time taken: 30.058 seconds'. The terminal window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The taskbar at the bottom shows several open windows, including 'Browsing HDF...' and 'cloudera@quic...'.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
File System Counters  
FILE: Number of bytes read=0  
FILE: Number of bytes written=683768  
FILE: Number of read operations=0  
FILE: Number of large read operations=0  
FILE: Number of write operations=0  
HDFS: Number of bytes read=2235  
HDFS: Number of bytes written=0  
HDFS: Number of read operations=19  
HDFS: Number of large read operations=0  
HDFS: Number of write operations=0  
Job Counters  
Launched map tasks=4  
Data-local map tasks=4  
Total time spent by all maps in occupied slots (ms)=102884  
Total time spent by all reduces in occupied slots (ms)=0  
Total time spent by all map tasks (ms)=102884  
Total vcore-milliseconds taken by all map tasks=102884  
Total megabyte-milliseconds taken by all map tasks=105353216  
Map-Reduce Framework  
Map input records=17  
Map output records=17  
Input split bytes=661  
Spilled Records=0  
Failed Shuffles=0  
Merged Map outputs=0  
GC time elapsed (ms)=624  
CPU time spent (ms)=6440  
Physical memory (bytes) snapshot=645132288  
Virtual memory (bytes) snapshot=6274723840  
Total committed heap usage (bytes)=423624704  
File Input Format Counters  
Bytes Read=0  
File Output Format Counters  
Bytes Written=0  
23/07/28 18:41:10 INFO mapreduce.ExportJobBase: Transferred 2.1826 KB in 54.1247 seconds (41.2935 bytes/sec)  
23/07/28 18:41:10 INFO mapreduce.ExportJobBase: Exported 17 records.  
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/hive_result_db --table result5 --username root --  
password cloudera --export-dir /user/cloudera/hive_dump --input-fields-terminated-by '|' --input-lines-terminated-by '\n';
```

Figure 29 shows syntax for successful export of hive_result5 into MySQL result5 table.

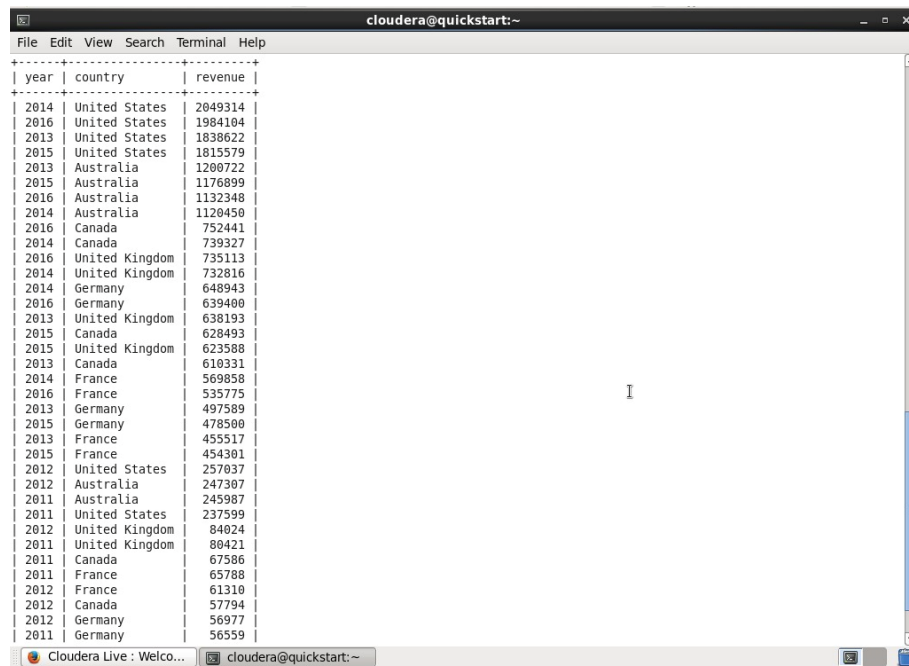
```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
+-----+  
| Germany | Bayern | 1426 |  
| Germany | Brandenburg | 198 |  
| Germany | Hamburg | 1836 |  
| Germany | Hessen | 2384 |  
| Germany | Nordrhein-Westfalen | 2484 |  
+-----+  
53 rows in set (0.00 sec)  
  
mysql> select * from result4;  
Empty set (0.00 sec)  
  
mysql> create table result5 (prod_cat varchar(50), sub_cat varchar(50), cost int, revenue int);  
Query OK, 0 rows affected (0.16 sec)  
  
mysql> select * from result5;  
+-----+-----+-----+-----+  
| prod_cat | sub_cat | cost | revenue |  
+-----+-----+-----+-----+  
| Accessories | Bike Racks | 213345 | 517800 |  
| Accessories | Bike Stands | 142140 | 344075 |  
| Accessories | Bottles and Cages | 598576 | 1409174 |  
| Accessories | Cleaners | 82930 | 198821 |  
| Clothing | Gloves | 64664 | 510744 |  
| Clothing | Jerseys | 311652 | 1157255 |  
| Clothing | Shorts | 125510 | 587195 |  
| Clothing | Socks | 10342 | 356459 |  
| Clothing | Vests | 66336 | 528655 |  
| Bikes | Mountain Bikes | 17477308 | 2105047 |  
| Bikes | Road Bikes | 25502270 | 3948528 |  
| Bikes | Touring Bikes | 6415790 | 2216878 |  
| Clothing | Caps | 470856 | 548777 |  
| Accessories | Fenders | 496819 | 1245733 |  
| Accessories | Helmets | 425530 | 2241977 |  
| Accessories | Hydration Packs | 417067 | 988750 |  
| Accessories | Tires and Tubes | 1943112 | 4670744 |  
+-----+-----+-----+-----+  
17 rows in set (0.01 sec)  
  
mysql>
```

Figure 30 shows the result of MySQL result5 table.

5. DATA INSIGHTS:

The data is a historical sales data of a company that sells different outdoor products which includes bike and its accessories, clothes, vest for both male and female. For this project, we have limited analysis to 5 queries to draw insight from product sales.

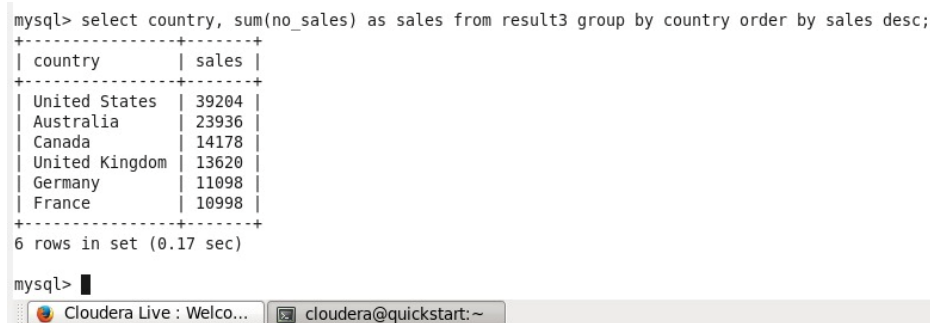
Data indicates sales to USA, Europe, and Australia. USA, being the world biggest market by GDP, recorded the biggest ever sales of outdoor products. In 2014, 2016 and 2013 were the strongest sales in the USA. Australia being the Darwin loves outdoor activities with favorable weather showed second strong sales. As shown in Figure 31, we can see the revenues by year.



The screenshot shows a terminal window titled 'cloudera@quickstart:~' with a table of sales data. The table has three columns: 'year', 'country', and 'revenue'. The data is sorted by year and country. The United States consistently shows the highest revenue, followed by Australia, Canada, United Kingdom, Germany, and France.

year	country	revenue
2014	United States	2849314
2016	United States	1984104
2013	United States	1838622
2015	United States	1815579
2013	Australia	1200722
2015	Australia	1176899
2016	Australia	1132348
2014	Australia	1120450
2016	Canada	752441
2014	Canada	739327
2016	United Kingdom	735113
2014	United Kingdom	732816
2014	Germany	648943
2016	Germany	639400
2013	United Kingdom	638193
2015	Canada	628493
2015	United Kingdom	623588
2013	Canada	610331
2014	France	569858
2016	France	535775
2013	Germany	497589
2015	Germany	478500
2013	France	455517
2015	France	454301
2012	United States	257037
2012	Australia	247907
2011	Australia	245987
2011	United States	237599
2012	United Kingdom	84024
2011	United Kingdom	80421
2011	Canada	67586
2011	France	65788
2012	France	61310
2012	Canada	57794
2012	Germany	56977
2011	Germany	56559

Figure 31 country with most revenue generated.



The screenshot shows a terminal window with a MySQL query result. The query is: `mysql> select country, sum(no_sales) as sales from result3 group by country order by sales desc;`. The result shows the top countries by total sales: United States (39204), Australia (23936), Canada (14178), United Kingdom (13620), Germany (11098), and France (10998).

```
mysql> select country, sum(no_sales) as sales from result3 group by country order by sales desc;
+-----+-----+
| country | sales |
+-----+-----+
| United States | 39204 |
| Australia | 23936 |
| Canada | 14178 |
| United Kingdom | 13620 |
| Germany | 11098 |
| France | 10998 |
+-----+-----+
6 rows in set (0.17 sec)

mysql>
```

Figure 32 USA and Australia is the biggest market for outdoor.

However, most of the sold products ordered were water bottles, patch kit and mountain tire tube. Which show strong outdoor activities because of the need to stay hydrated and repairs of already bought bikes. It is important to monitor the inventory of accessories so that accessories are available for customers. Figure 33 shows the products bought by customers.



prod	orders
Water Bottle - 30 oz.	164086
Patch Kit/8 Patches	157583
Mountain Tire Tube	102792
Sport-100 Helmet	70178
AWC Logo Cap	67316
Road Tire Tube	62296
Fender Set - Mountain	62118
Touring Tire Tube	56802
Road Bottle Cage	40164
Road-150 Red	39144
Mountain Bottle Cage	37480
Touring-1000 Yellow	32846
Road-750 Black	29140
Mountain-200 Black	27604
Bike Wash - Dissolver	27579
HL Mountain Tire	27562
Touring-1000 Blue	27416
LL Road Tire	26584
Mountain-200 Silver	26330
Road-550-W Yellow	25918
Classic Vest	24868
Women's Mountain Shorts	23452
ML Mountain Tire	20992
ML Road Tire	20865
Touring-2000 Blue	20480
Hydration Pack - 70 oz.	19857
Long-Sleeve Logo Jersey	19620
Short-Sleeve Classic Jersey	18732
Road-350-W Yellow	17346
Road-250 Black	16632
HL Road Tire	15610
Racing Socks	14458
Half-Finger Gloves	13874
Touring-3000 Yellow	12830
LL Mountain Tire	12744
Touring-3000 Blue	11606

Figure 33 most orders products.

Without doubt we can confirm the countries with the most revenue and we can see in figure 33 that the top 10 products are predominantly accessories. So, with confirmation from figure 34, we confirm that accessories are the leading source of revenue for this company.

Therefore, we recommend that the company further analysis their inventory to maintain enough products for their customer especially towards inventory limits. Secondly, the company can increase their advertising campaign to drive sales of other products, ask for product feedback to drive seasonal growth further higher.

6. CONCLUSION:

From the analysis we can see that the company sales seasonal products which generates good sales. Their biggest market is USA and Australia, so the company must continue run targeted advert to improve sales, especially before the beginning of the season and expand their advert promoting healthy outdoor activities.

7. CHALLENGES:

While transferring our Hive result data from HDFS to MySQL, we noticed that our query would run successfully but the table in MySQL is empty. After further investigation, we noticed we did not include the right delimiter causing our not be clean enough for transfer. After further investigation, we found the right syntax that removes the pipe '|' thereby making our data to be successfully transferred from HDFS to MySQL. It is a lesson learned. Figure 35 is what our data looks like without removing the pipe delimiter.


```

cloudera@quickstart:~/Downloads
File Edit View Search Terminal Help
File Output Format Counters
Bytes Written=0
23/07/28 18:50:43 INFO mapreduce.ExportJobBase: Transferred 2.2866 MB in 59.0824 seconds (39.6301 KB/sec)
23/07/28 18:50:43 INFO mapreduce.ExportJobBase: Exported 113034 records.
[cloudera@quickstart ~]$ cd Downloads
[cloudera@quickstart Downloads]$ ls
0000000 0 0000000 0(1)
[cloudera@quickstart Downloads]$ more 0000000_0
AustraliaNew South Wales10412
AustraliaQueensland5220
AustraliaSouth Australia1564
AustraliaTasmania724
AustraliaVictoria6016
CanadaAlberta56
CanadaBritish Columbia14116
CanadaOntario6
FranceCharente-Maritime148
FranceEssonne994
FranceGaronne (Haute)208
FranceHauts de Seine1084
FranceLoir et Cher120
FranceLoiret302
FranceMoselle306
FranceNord1070
FrancePas de Calais90
FranceSeine (Paris)328
FranceSeine Saint Denis1684
FranceSeine et Marne394
FranceSomme134
FranceVal d'Oise264
FranceVal de Marne158
FranceYveline354
GermanyBayern1426
GermanyBrandenburg198
GermanyHamburg1836
GermanyHessen1384
GermanyNordrhein-Westfalen484
GermanySaarland778
United KingdomEngland13620

```

Figure 34 shows hive-result table result with unknown delimiter.

8. PROJECT PARTICIPANTS:

PROJECT PARTICIPANTS	STUDENT ID
Student Name: Mahima Akula	C0908140
Student Name: Modupeola Omodunni Oyatokun	C0895705
Student Name: Jumoke Yekeen	C0900481
Student Name: Chibuike Okoroama	C0892150
Student Name: Diksha	C0908141
Student Name: Harish Kundal	C0906990