

**PROJECT DESCRIPTION:-** The Project aims to display Vehicle Sales Data of Different Countries of the World. The dataset contains various attributes of the Sales data ranging from year 2003 to 2005. The Dataset is stored in the form of CSV file and the project aims to stage it to HDFS and further analysis is to be performed using HIVE.

**DESCRIPTION OF DATASET:-** The dataset contains almost 2900 rows and 21 columns(attributes) about the vehicle sales data. The Unique attributes of the dataset and their data-types are :

- ORDERNUMBER int,
- QUANTITYORDERED int,
- PRICEEACH float,
- ORDERLINENUMBER int,
- SALES float,
- STATUS string,
- QTR\_ID int,
- MONTH\_ID int,
- YEAR\_ID int,
- PRODUCTLINE string,
- MSRP int,
- PRODUCTCODE string,
- PHONE string,
- CITY string,
- STATE string,
- POSTALCODE string,
- COUNTRY string,
- TERRITORY string,
- CONTACTLASTNAME string,
- CONTACTFIRSTNAME string,
- DEALSIZE string

2824 lines (2824 xloc) | 352 KB

Raw Blame

Search this file...

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	STATUS	QTR_ID	MONTH_ID	YEAR_ID	PRODUCTLINE	MSRP	PRODUCTCODE	PHONE	CITY
1	10107	30	95.7	2	2871	Shipped	1	2	2003	Motorcycles	95	S10_1678	2125557818	NYC
2	10121	34	81.35	5	2765.9	Shipped	2	5	2003	Motorcycles	95	S10_1678	26.47.1555	Reims
3	10134	41	94.74	2	3884.34	Shipped	3	7	2003	Motorcycles	95	S10_1678	+33 1 46 62 7555	Paris
4	10145	45	83.26	6	3746.7	Shipped	3	8	2003	Motorcycles	95	S10_1678	6265557265	Pasadena
5	10159	49	100	14	5205.27	Shipped	4	10	2003	Motorcycles	95	S10_1678	6505551386	San Francisco
6	10168	36	96.66	1	3479.76	Shipped	4	10	2003	Motorcycles	95	S10_1678	6505556809	Burlingame
7	10180	29	86.13	9	2497.77	Shipped	4	11	2003	Motorcycles	95	S10_1678	20.16.1555	Lille
8	10188	48	100	1	5512.32	Shipped	4	11	2003	Motorcycles	95	S10_1678	+47 2267 3215	Bergen
9	10201	22	98.57	2	2168.54	Shipped	4	12	2003	Motorcycles	95	S10_1678	6505555787	San Francisco
10	10211	41	100	14	4708.44	Shipped	1	1	2004	Motorcycles	95	S10_1678	(1) 47.55.6555	Paris
11	10223	37	100	1	3965.66	Shipped	1	2	2004	Motorcycles	95	S10_1678	03 9520 4555	Melbourne
12	10237	23	100	7	2333.12	Shipped	2	4	2004	Motorcycles	95	S10_1678	2125551500	NYC
13	10251	28	100	2	3188.64	Shipped	2	5	2004	Motorcycles	95	S10_1678	2015559350	Newark
14	10263	34	100	2	3676.76	Shipped	2	6	2004	Motorcycles	95	S10_1678	2035552570	Bridgewater

The glimpse of the dataset is shown above.

## **PROBLEM STATEMENT:-**

- Store raw data into hdfs location
- Create a internal hive table "sales\_order\_csv" which will store csv data sales\_order\_csv .. make sure to skip header row while creating table
- Load data from hdfs path into "sales\_order\_csv"
- Create an internal hive table which will store data in ORC format "sales\_order\_orc"
- Load data from "sales\_order\_csv" into "sales\_order\_orc".

Placing the given Dataset in the HDFS:- Step1 - Create a Directory in HDFS, ensuring all the daemons are started.

**hadoop fs -mkdir /user/hive/project**

Step2 - Copy the Dataset from local to HDFS directory.

**hadoop fs -copyFromLocal /home/cloudera/Downloads/Sales\_order\_data.csv /user/hive/project/sales\_order\_data.csv**

**Creating The Database:** Create a database named 'hive\_assignment', and use it for further purpose.

- Create database hive\_assignment;
- Use hive\_assignment;

**Creating the internal hive table :** After creating database , we have to create an internal table.

```
create table sales_order_data_csv
(ORDERNUMBER int, QUANTITYORDERED int,
PRICEEACH float, ORDERLINENUMBER int,
SALES float, STATUS string,
QTR_ID int, MONTH_ID int,
YEAR_ID int, PRODUCTLINE string,
MSRP int, PRODUCTCODE string,
PHONE string, CITY string,
STATE string, POSTALCODE string,
COUNTRY string, TERRITORY string,
CONTACTLASTNAME string,
CONTACTFIRSTNAME string, DEALSIZE string)
row format delimited
fields terminated by ','
tblproperties("skip.header.line.count"="1");
```



```
hive> create table sales_order_data_csv(
> ORDERNUMBER int,
> QUANTITYORDERED int,
> PRICEEACH float,
> ORDERLINENUMBER int,
> SALES float,
> STATUS string,
> QTR_ID int,
> MONTH_ID int,
> YEAR_ID int,
> PRODUCTLINE string,
> MSRP int,
> PRODUCTCODE string,
> PHONE string,
> CITY string,
> STATE string,
> POSTALCODE string,
> COUNTRY string,
> TERRITORY string,
> CONTACTLASTNAME string,
> CONTACTFIRSTNAME string,
> DEALSIZE string)
> row format delimited
> fields terminated by ','
> tblproperties("skip.header.line.count"="1");
OK
Time taken: 2.41 seconds
hive>
```

### Creating Internal table

**Loading data from HDFS path to the table created:** After creating the table sales\_order\_csv, we have to load the data content inside it.

load data local inpath '/home/cloudera/Downloads/sales\_order\_data.csv' into table sales\_order\_data.csv;

Create an internal hive table which will store data in ORC format "sales\_order\_orc" :

create table sales\_data\_orc  
(ORDERNUMBER int, QUANTITYORDERED int,  
PRICEEACH float, ORDERLINENUMBER int,  
SALES float, STATUS string,  
QTR\_ID int, MONTH\_ID int,  
YEAR\_ID int, PRODUCTLINE string,  
MSRP int, PRODUCTCODE string,  
PHONE string, CITY string,  
STATE string, POSTALCODE string,  
COUNTRY string, TERRITORY string,  
CONTACTLASTNAME string,  
CONTACTFIRSTNAME string, DEALSIZE string)  
Stored as orc;

Load data from "sales\_order\_csv" into "sales\_order\_orc": Now loading the content from csv table to orc table.

From sales\_order\_data.csv insert overwrite table sales\_data\_orc select \*;



```
cloudera-quickstart-vm-5.13.0-0-virtualbox [Running] - Oracle VM VirtualBox
File Machine View Search Terminal Help
Applications Places System
cloudera@quickstart:~$
cloudera@quickstart:~$ hive> load data local inpath '/home/cloudera/Downloads/sales_order_data.csv' into table sales_data.csv;
Loading data to table hive_assignment.sales_data.csv
Table hive_assignment.sales_data.csv stats: [numFiles=1, totalSize=360233]
OK
Time taken: 3.851 seconds
hive> create table sales_data_orc
> (ORDERNUMBER int,
> QUANTITYORDERED int,
> PRICEEACH float,
> ORDERLINENUMBER int,
> SALES float,
> STATUS string,
> QTR_ID int,
> MONTH_ID int,
> YEAR_ID int,
> PRODUCTLINE string,
> MSRP int,
> PRODUCTCODE string,
> PHONE string,
> CITY string,
> STATE string,
> POSTALCODE string,
> COUNTRY string,
> TERRITORY string,
> CONTACTLASTNAME string,
> CONTACTFIRSTNAME string,
> DEALSIZE string)
> stored as orc;
OK
Time taken: 0.889 seconds
hive> from sales_data.csv insert overwrite table sales_data_orc select *;
Query ID = cloudera_26228927075252_8e766205-d75b-4878-a7d6-14ed31e5f7ab
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1664284577728_0001, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1664284577728_0001/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1664284577728_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2022-09-27 07:53:32,240 Stage-1 map = 0%, reduce = 0%
2022-09-27 07:53:35,839 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 7.8 sec
MapReduce Total cumulative CPU time: 7 seconds 800 msec
Ended Job = job_1664284577728_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to: hdfs://quickstart.cloudera:8020/user/hive/warehouse/hive_assignment.db/sales_data_orc/.hive-staging_hive_2022-09-27_07-52-35_747_8749415768988824927-1/-ext-10000
Loading data to table hive_assignment.sales_data_orc
Table hive_assignment.sales_data_orc stats: [numFiles=1, numRows=2823, totalSize=37548, rawDataSize=3153291]
[ Hive-Class/hive.class... ] [ Downloads ] [ cloudera@quickstart:~ ] [ cloudera@quickstart:~ ]
```

The above figure depicts the successful loading of data from csv table to sales\_data\_orc table. Now that the data is loaded in Hive table, further analysis is to be performed in Hive shell.

## **PROBLEM SCENARIO:**

Below are some problem statements that are to be executed using Hive commands on the sales\_orc table.

### **1) Calculate total sales per year.**

**Select sum(sales) , year\_id from sales\_data\_orc  
group by year\_id;**

```

Time taken: 89.24 seconds
hive> select sum(sales), year_id from sales_data_orc group by year_id;
Query ID = cloudera_20220927075555_dc29e78a-6500-4037-0872-6d7d1f337305
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.reducers=<number>
Starting Job = job_1664284577728_0002, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1664284577728_0002/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1664284577728_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-09-27 07:56:22,073 Stage-1 map = 0%, reduce = 0%
2022-09-27 07:56:42,352 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 5.0 sec
2022-09-27 07:57:00,577 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.15 sec
MapReduce Total cumulative CPU time: 9 seconds 150 msec
Finished Job = job_1664284577728_0002
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1; Reduce: 1; Cumulative CPU: 9.15 sec HDFS Read: 36985 HDFS Write: 70 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 150 msec
OK
3516979.547241211      2003
47224162.593383789    2004
1791486.7086791982    2005
Time taken: 63.59 seconds, Fetched: 3 row(s)
hive> set hive.cli.print.header=true;

```

After running the above query , we got the desired output of total sales year wise.

Total Sales	Year
3516979.5472	2003
47224162.5933	2004
1791486.7086	2005

From the output, it can be concluded that the year 2004 had the highest amount of sales in comparison to other two years.

### **2) Find a product for which maximum orders were placed.**

**Select PRODUCTLINE, Count(QUANTITYORDERED) as  
count From sales\_data\_orc  
Group by PRODUCTLINE  
Order by count Desc limit 3;**

After running the above query , the output is:

PRODUCTLINE	MAXIMUM
Classic Cars	967
Vintage Cars	607
Motorcycles	331

```

2823
Time taken: 125.287 seconds, Fetched: 1 row(s)
hive> select PRODUCTLINE,Count(QUANTITYORDERED) as count from sales_data_orc group by PRODUCTLINE order by count DESC Limit 3;
Query ID = cloudera_20221007204141_b4607923-c250-4050-b080-a305a7180803
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_1665199560964_0002, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0002/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0002
Hadoop job information for Stage:1: number of mappers: 1; number of reducers: 1
2022-10-07 20:41:41.768 Stage:1 map = 0%, reduce = 0%
2022-10-07 20:42:09.950 Stage:1 map = 100%, reduce = 0%, Cumulative CPU 6.38 sec
2022-10-07 20:42:27.170 Stage:1 map = 100%, reduce = 100%, Cumulative CPU 10.33 sec
MapReduce Total cumulative CPU time: 10 seconds 330 msec
Ended Job = job_1665199560964_0002
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_1665199560964_0003, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0003
Hadoop job information for Stage:2: number of mappers: 1; number of reducers: 1
2022-10-07 20:42:47.254 Stage:2 map = 0%, reduce = 0%
2022-10-07 20:42:59.710 Stage:2 map = 100%, reduce = 0%, Cumulative CPU 2.79 sec
2022-10-07 20:43:14.347 Stage:2 map = 100%, reduce = 100%, Cumulative CPU 6.53 sec
MapReduce Total cumulative CPU time: 6 seconds 530 msec
Ended Job = job_1665199560964_0003
MapReduce Jobs Launched:
Stage:Stage:1: Map: 1 Reduce: 1 Cumulative CPU: 10.33 sec HDFS Read: 28450 HDFS Write: 308 SUCCESS
Stage:Stage:2: Map: 1 Reduce: 1 Cumulative CPU: 6.53 sec HDFS Read: 5388 HDFS Write: 56 SUCCESS
Total MapReduce CPU Time Spent: 16 seconds 608 msec
OK
Classic Cars      967
Vintage Cars      607
Motorcycles       331
Time taken: 121.484 seconds, Fetched: 3 row(s)
hive>

```

Here, it can be concluded that Classic cars were sold mostly in the year of 2003 - 2005 across different parts of the world.

### 3) Calculate Total sales for each quarter.

**Select sum(SALES) as Total, QTR\_ID from sales\_data\_orc Group by QTR\_ID;**

```

Time taken: 48.583 seconds, Fetched: 4 row(s)
hive> select sum(SALES) as Total,QTR_ID from sales_data_orc group by QTR_ID;
Query ID = cloudera_20220927081519_80c143ad-7644-428e-b4ce-c063c8c1b24b
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_1664284577728_0006, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1664284577728_0006/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1664284577728_0006
Hadoop job information for Stage:1: number of mappers: 1; number of reducers: 1
2022-09-27 08:16:14.141 Stage:1 map = 0%, reduce = 0%
2022-09-27 08:16:28.634 Stage:1 map = 100%, reduce = 0%, Cumulative CPU 5.02 sec
2022-09-27 08:16:44.572 Stage:1 map = 100%, reduce = 100%, Cumulative CPU 9.12 sec
MapReduce Total cumulative CPU time: 9 seconds 120 msec
Ended Job = job_1664284577728_0006
MapReduce Jobs Launched:
Stage:Stage:1: Map: 1 Reduce: 1 Cumulative CPU: 9.12 sec HDFS Read: 37249 HDFS Write: 81 SUCCESS
Total MapReduce CPU Time Spent: 9 seconds 120 msec
OK
Total    qtr_id
2358817 726501465    1
2468120 3637174805   2
1738910 808959961    3
1974780 010925293    4
Time taken: 48.583 seconds, Fetched: 4 row(s)
hive>

```

The result of the above query is shown in the above image. It can be concluded that the last quarter\_id has the most amount of sales.

Total	Qtr_Id
2350817.726	1
2048120.302	2
1758910.808	3
3874780.010	4

**4) In which quarter sales was minimum:**

**Select sum(SALES) as Sum, QTR\_ID, YEAR\_ID  
from sales\_data\_orc  
Group by QTR\_ID, YEAR\_ID  
Order by sum Asc limit 3;**

```

Motorcycles 331
Time taken: 121.484 seconds, Fetched: 3 row(s)
hive> select sum(sales) as sum, qtr_id, year_id from sales_data_orc group by year_id, qtr_id order by total asc limit 3;
Query ID = cloudera_20221007210808_4ba1b06c-cb17-46f5-990b-fa754ab05b3b
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_1665199560964_0004, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0004/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0004
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-10-07 21:09:26,187 Stage-1 map = 0%, reduce = 0%
2022-10-07 21:09:49,672 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.57 sec
2022-10-07 21:10:05,434 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 8.32 sec
MapReduce Total cumulative CPU time: 8 seconds 320 msec
Ended Job = job_1665199560964_0004
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_1665199560964_0005, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0005/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0005
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-10-07 21:10:23,359 Stage-2 map = 0%, reduce = 0%
2022-10-07 21:10:36,065 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 4.09 sec
2022-10-07 21:10:50,592 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 8.74 sec
MapReduce Total cumulative CPU time: 8 seconds 740 msec
Ended Job = job_1665199560964_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 8.32 sec HDFS Read: 36993 HDFS Write: 386 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 8.74 sec HDFS Read: 5534 HDFS Write: 75 SUCCESS
Total MapReduce CPU Time Spent: 17 seconds 60 msec
OK
445094.6897583808 2003 1
502365.2218817578 2003 2
649514.5418819862 2003 3
Time taken: 114.285 seconds, Fetched: 3 row(s)
hive>

```

From the above output, Quarter Id 1 in year 2003 has the minimum amount of Sales recorded of value “445094.68”.

**5) In which Country Sales was Maximum and in which country sales was minimum.**

**Select max(SALES) as Min\_Max, COUNTRY  
From Sales\_data\_orc group by COUNTRY  
Order by Min\_Max DESC limit 1  
UNION ALL  
Select Min(SALES) as Min\_Max, COUNTRY  
From Sales-data\_orc group by COUNTRY  
Order by Min\_Max ASC limit 1;**

```
649514-5415839862      2003      3
Time Taken: 114.293 seconds, Fetched: 3 row(s)
hive> select COUNTRY, max(SALES) as max_min from sales_data_orc group by COUNTRY order by max_min desc limit 1
+-----+
> select COUNTRY, min(SALES) as max_min from sales_data_orc group by COUNTRY order by max_min asc limit 1;
Query ID = cloudera.20221008034840_dbc1868-9505-470a-93a9-cf4a8bea6678
Total jobs = 5
Launching Job 1 out of 5
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_1665199560964_0006, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0006/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0006
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-10-08 03:40:30,009 Stage-1 map = 0%, reduce = 0%
2022-10-08 03:40:48,940 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.63 sec
2022-10-08 03:40:48,927 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.54 sec
MapReduce Total cumulative CPU time: 9 seconds 540 msec
Ended Job = job_1665199560964_0006
Launching Job 2 out of 5
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_1665199560964_0007, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0007/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0007
Hadoop job information for Stage-4: number of mappers: 1; number of reducers: 1
2022-10-08 03:40:24,612 Stage-4 map = 0%, reduce = 0%
2022-10-08 03:40:39,448 Stage-4 map = 100%, reduce = 0%, Cumulative CPU 4.22 sec
2022-10-08 03:40:57,491 Stage-4 map = 100%, reduce = 100%, Cumulative CPU 8.84 sec
MapReduce Total cumulative CPU time: 8 seconds 840 msec
Ended Job = job_1665199560964_0007
Launching Job 3 out of 5
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_1665199560964_0010, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0010/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0010
Hadoop job information for Stage-3: number of mappers: 2; number of reducers: 0
2022-10-08 03:52:43,968 Stage-3 map = 0%, reduce = 0%
2022-10-08 03:52:24,439 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 9.71 sec
MapReduce Total cumulative CPU time: 9 seconds 710 msec
Ended Job = job_1665199560964_0010
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.54 sec HDFS Read: 36890 HDFS Write: 640 SUCCESS
Stage-Stage-4: Map: 1 Reduce: 1 Cumulative CPU: 8.84 sec HDFS Read: 36897 HDFS Write: 640 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 7.89 sec HDFS Read: 4815 HDFS Write: 121 SUCCESS
Stage-Stage-5: Map: 1 Reduce: 1 Cumulative CPU: 9.88 sec HDFS Read: 4819 HDFS Write: 124 SUCCESS
Stage-Stage-3: Map: 2 Cumulative CPU: 9.71 sec HDFS Read: 6325 HDFS Write: 26 SUCCESS
Total MapReduce CPU Time Spent: 45 seconds 800 msec
OK
France 482.13
USA 14082.8
Time Taken: 257.081 seconds, Fetched: 2 row(s)
hive>
```

```
MapReduce Total cumulative CPU time: 7 seconds 690 msec
Ended Job = job_1665199560964_0011
Launching Job 2 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_1665199560964_0012, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0012/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0012
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-10-08 04:01:11,646 Stage-2 map = 0%, reduce = 0%
2022-10-08 04:01:24,441 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 3.85 sec
2022-10-08 04:01:41,167 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 8.43 sec
MapReduce Total cumulative CPU time: 8 seconds 430 msec
Ended Job = job_1665199560964_0012
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.69 sec HDFS Read: 38352 HDFS Write: 6528 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 8.43 sec HDFS Read: 11876 HDFS Write: 137 SUCCESS
Total MapReduce CPU Time Spent: 16 seconds 128 msec
OK
337688.4899291902      Madrid 1
139588.05113895664      Madrid 2
215580.80963134766      Madrid 4
180611.0999311719       NYC 4
267315.2586669922       San Rafael 1
Time Taken: 183.739 seconds, Fetched: 5 row(s)
hive>
```

The above two image depicts the final output. The Country of **USA** has the **maximum** amount of Sales of value “14082.8” while the Country of **France** has **lowest** amount of Sales recorded which was “482.3”.

## 6) Calculate Quarterly Sales For each City.

```
Time taken: 257.083 seconds, Fetched: 2 row(s)
hive> select Sum(SALES) as Total, CITY_QTR_ID from sales_data_orc group by CITY_QTR_ID order by Total DESC limit 5;
Query ID = cloudera.20221008035959_343a6eaf-9208-485e-9054-65db358124fa
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_1665199560964_0011, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0011/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0011
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-10-08 04:00:25,088 Stage-1 map = 0%, reduce = 0%
2022-10-08 04:00:40,094 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.64 sec
2022-10-08 04:00:53,463 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 7.69 sec
MapReduce Total cumulative CPU time: 7 seconds 690 msec
Ended Job = job_1665199560964_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_1665199560964_0012, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0012/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0012
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-10-08 04:01:11,646 Stage-2 map = 0%, reduce = 0%
2022-10-08 04:01:24,441 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 3.85 sec
2022-10-08 04:01:41,167 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 8.43 sec
MapReduce Total cumulative CPU time: 8 seconds 430 msec
Ended Job = job_1665199560964_0012
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.69 sec HDFS Read: 38352 HDFS Write: 6528 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 8.43 sec HDFS Read: 11876 HDFS Write: 137 SUCCESS
Total MapReduce CPU Time Spent: 16 seconds 128 msec
OK
337688.4899291902      Madrid 1
139588.05113895664      Madrid 2
215580.80963134766      Madrid 4
180611.0999311719       NYC 4
267315.2586669922       San Rafael 1
Time Taken: 183.739 seconds, Fetched: 5 row(s)
hive>
```

Select Sum(SALES) as Total , City,QTR\_ID  
From sales\_data\_orc  
Group by City,QTR\_ID  
Order by Total Desc Limit 5;



**The Top 5 sales city-wise recorded were:**

Total	City	QTR_ID
357668.48	Madrid	1
339588.05	Madrid	2
315580.80	Madrid	4
300011.69	NYC	4
267315.25	San Rafael	1

**7) Find the Month for each year in which maximum number of Quantities were sold.**

Select MONTH\_ID, Count(QUANTITYORDERED) as Count  
From Sales\_data\_orc  
Group by MONTH\_ID  
Order by Count Desc limit 5;

```
hive> set hive.cli.print.header=true;
hive> select month_id, count(QUANTITYORDERED) as Count from Sales_data_orc group by MONTH_ID order by Count Desc limit 5;
Query ID = cloudera-2022100804111.70977897-3caf-420d-a215-c86feb2d25a0
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.reducers=number
Starting Job = job_1665199560964_0015, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0015/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0015
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-10-08 04:12:14.162 Stage-1 map = 0%, reduce = 0%
2022-10-08 04:12:29.026 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 5.29 sec
2022-10-08 04:12:48.419 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 9.39 sec
MapReduce Total cumulative CPU time: 9 seconds 390 msec
Ended Job = job_1665199560964_0015
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.reducers=number
Starting Job = job_1665199560964_0016, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1665199560964_0016/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1665199560964_0016
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-10-08 04:13:07.176 Stage-2 map = 0%, reduce = 0%
2022-10-08 04:13:19.223 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 2.71 sec
2022-10-08 04:13:31.461 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 6.66 sec
MapReduce Total cumulative CPU time: 6 seconds 660 msec
Ended Job = job_1665199560964_0016
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 9.39 sec HDFS Read: 28920 HDFS Write: 338 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 6.66 sec HDFS Read: 5397 HDFS Write: 32 SUCCESS
Total MapReduce CPU Time Spent: 16 seconds 50 msec
OK
month_id      count
11            597
10            317
5             252
1             229
2             224
Time taken: 95.759 seconds, Fetched: 5 row(s)
```

Month id	Count
11	597
10	317
5	252
1	229
2	224

## 8) Show Distinct Countries from the dataset

```
268944.6882368164 Paris
Time taken: 22.198 seconds, Fetched: 5 row(s)
hive> select distinct(country) from sales_data_orc;
Query ID = cloudera_20220928084040_5f3649a6-ed10-4edd-88fc-a215e2e945c5
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.reducers=<number>
Starting Job = job_1664360922326_0011, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1664360922326_0011/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1664360922326_0011
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-09-28 08:40:44.242 Stage-1 map = 0%, reduce = 0%
2022-09-28 08:40:53.688 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.72 sec
2022-09-28 08:41:01.168 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.74 sec
MapReduce Total cumulative CPU time: 3 seconds 740 msec
Ended Job = job_1664360922326_0011
MapReduce Jobs Launched:
Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.74 sec HDFS Read: 27846 HDFS Write: 145 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 740 msec
OK
country
Australia
Austria
Belgium
Canada
Denmark
Finland
France
Germany
Ireland
Italy
Japan
Norway
Philippines
Singapore
Spain
Sweden
Switzerland
UK
USA
Time taken: 27.985 seconds, Fetched: 19 row(s)
hive>
```