HIVE ASSIGNMENT 1

Table of Contents

Download vehicle sales data

Downloaded comma separated file “sales_order_data.csv”
https://github.com/shashank-mishra219/Hive-Class/blob/main/sales_order_data.csv

Storing raw data into hdfs location

Copied csv file "sales_order_data.csv" into local Cloudera folder using FileZilla.
Then copied the file from local to HDFS location

```
# CREATE A DIRECTORY IN HDFS
hadoop fs -mkdir /user/cloudera/sourav/sales

# COPY THE FILE FROM LOCAL TO HDFS
hadoop fs -copyFromLocal /home/cloudera/sourav/data/sales_order_data.csv /user/cloudera/sourav/sales
```

Creating internal hive table

Created an internal hive table "sales_order_csv" which will store csv data sales_order_csv
Note: Skip header row while creating table

```
# CONNECT TO HIVE
hive

# CREATE HIVE DATABASE
create database hive_class_b1;
use hive_class_b1;

# CREATE INTERNAL HIVE TABLE IN CSV FORMAT
create table sales_order_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");
```

Loading data from hdfs into sales_order_data_csv table

```
load data inpath '/user/cloudera/sourav/sales/' into table sales_order_csv;
```

Creating internal hive table in ORC format

```
# CREATE INTERNAL HIVE TABLE IN ORC FORMAT
create table sales_order_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 internal CSV table into ORC table

```
from sales_order_csv insert overwrite table sales_order_orc select *;
```

Display first 10 rows from the ORC table

```
# Display column name
set hive.cli.print.header = true;

# Display first 10 rows from sales_order_data_orc table
select * from sales_order_data_orc limit 10;
```

Perform queries on ORC table

(a.) Calculate total sales per year

```
select year_id, sum(sales) as total_sales from sales_order_orc group by year_id;
```

```
year_id total_sales
2003    3516979.547241211
2004    4724162.593383789
2005    1791486.7086791992
```

(b.) Find a product for which maximum orders were placed

```
select p.productline as product, p.ord_qty as max_order
from (
select productline, sum(quantityordered) as ord_qty
from sales_order_orc
group by productline
) p
order by max_order desc
limit 1;
```

```
product max_order
Classic Cars    33992
```

(c.) Calculate the total sales for each quarter

```
select YEAR_ID as YEAR,QTR_ID AS QUARTER,sum(SALES) AS TOT_SALES
from sales_order_orc
group by YEAR_ID,QTR_ID;
```

```
year    quarter tot_sales
2003    1       445094.6897583008
2003    2       562365.2218017578
2003    3       649514.5415039062
2003    4       1860005.094177246
2004    1       833730.6786499023
2004    2       766260.7305297852
2004    3       1109396.2674560547
2004    4       2014774.9167480469
2005    1       1071992.3580932617
2005    2       719494.3505859375
```

(d.) In which quarter sales was minimum

```
select YEAR_ID as YEAR,QTR_ID AS QUARTER,sum(SALES) AS MINIMUM_SALES
from sales_order_orc
group by YEAR_ID,QTR_ID
order by MINIMUM_SALES limit 1;
```

```
year    quarter minimum_sales
2003    1       445094.6897583008
```

(e.) In which country sales was maximum and in which country sales was minimum

```
WITH country_tbl as(
select COUNTRY,sum(SALES) AS TOT_SALES
from sales_order_orc
group by COUNTRY),
min_max_tbl as (
select max(TOT_SALES) MAX_SALES,min(TOT_SALES) MIN_SALES
from country_tbl)
select t1.COUNTRY AS COUNTRY,t1.TOT_SALES AS SALES,
(case
when t1.TOT_SALES=t2.MAX_SALES then 'MAX SALES'
when t1.TOT_SALES=t2.MIN_SALES then 'MIN SALES'
else ''
end) as MIN_MAX
from country_tbl t1
left join min_max_tbl t2
where t1.TOT_SALES=t2.MAX_SALES OR t1.TOT_SALES=t2.MIN_SALES;
```

country	sales	min_max
Ireland	57756.43029785156	MIN SALES
USA	3627982.825744629	MAX SALES

(f.) Calculate quarterly sales for each city

```
select CITY,QTR_ID AS QUARTER,sum(SALES) AS TOT_SALES
from sales_order_orc
group by CITY,QTR_ID;
```

city	quarter	tot_sales
Aaarhus	4	100595.5498046875
Allentown	2	6166.7998046875
Allentown	3	71930.61041259766
Allentown	4	44040.729736328125
Barcelona	2	4219.2001953125
Barcelona	4	74192.66003417969
Bergamo	1	56181.320068359375
Bergamo	4	81774.40008544922
Bergen	3	16363.099975585938
Bergen	4	95277.17993164062

------ (continue) Fetched : 182 rows

(h.) Find a month for each year in which maximum number of quantities were sold

```
with sales_by_year_month as(
select YEAR_ID,MONTH_ID,sum(quantityordered) as ord_qty
from sales_order_orc
group by YEAR_ID,MONTH_ID),
max_sales_by_year as(
select YEAR_ID,max(ord_qty) as max_ord_qty
from sales_by_year_month
group by YEAR_ID)
select t1.YEAR_ID,t1.MONTH_ID,t2.max_ord_qty
from sales_by_year_month t1
left join max_sales_by_year t2
on t1.YEAR_ID=t2.YEAR_ID and t1.ord_qty=t2.max_ord_qty
where t2.max_ord_qty is not null;
```

```
t1.year_id      t1.month_id     t2.max_ord_qty
2003    11      10179
2004    11      10678
2005    5       4357
Time taken: 66.053 seconds, Fetched: 3 row(s)
```