

## Questions

1. Download vehicle sales data ->  
[https://github.com/shashank-mishra219/Hive-Class/blob/main/sales\\_order\\_data.csv](https://github.com/shashank-mishra219/Hive-Class/blob/main/sales_order_data.csv) Done
2. Store raw data into hdfs location Done
3. Create an internal hive table "sales\_order\_csv" which will store csv data sales\_order\_csv .. make sure to skip header row while creating table

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

4. Load data from hdfs path into "sales\_order\_csv"

```
hive> load data inpath '/tmp/vehicle_sales_order_data.csv' into table
vehicle_sales_order_csv;
```

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

```
hive> create table vehicle_sales_order_orc(
  > ORDERNUMBER int,QUANTITYORDERED int,PRICEEACH int,ORDERLINENUMBER
int,SALES int,STATUS string,QTR_ID int,MONTH_ID int,YEAR_ID int,PRODUCTLINE
string,MSRP int,PRODUCTCODE string,PHONE int,CITY string,STATE
string,POSTALCODE int,COUNTRY string,TERRITORY string,CONTACTLASTNAME
string,CONTACTFIRSTNAME string,DEALSIZE string
  > )
  > stored as orc;
```

6. Load data from "sales\_order\_csv" into "sales\_order\_orc"

```
hive> from vehicle_sales_order_csv insert overwrite table  
vehicle_sales_order_orc select *;
```

Perform below mentioned queries on "sales\_order\_orc" table :

1. Calculate total sales per year
2. Find a product for which maximum orders were placed
3. Calculate the total sales for each quarter
4. In which quarter sales was minimum
5. In which country sales was maximum and in which country sales was minimum
6. Calculate quarterly sales for each city
7. Find a month for each year in which maximum number of quantities were sold

**Solutions:**

**1-Calculate total sales per year**

```
hive> set hive.cli.print.header=True;  
hive> select year_id,sum(sales) as total_sales from  
    > vehicle_sales_order_orc  
    > group by year_id;  
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the  
future versions. Consider using a different execution engine (i.e. spark,  
tez) or using Hive 1.X releases.  
Query ID = root_20230304195120_b63049a7-8e1c-4e8a-945d-676dafa22808  
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>
Job running in-process (local Hadoop)
2023-03-04 19:51:22,335 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local635483348_0003
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 996066 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
year_id total_sales
2003      3516514
2004      4723531
2005      1791264
Time taken: 1.965 seconds, Fetched: 3 row(s)
hive>

```

## 2-Find a product for which maximum orders were placed

```

hive> with temp as (
  > select productline,sum(quantityordered) as qo
  > from vehicle_sales_order_orc
  > group by productline)
  > select * from temp order by qo desc limit 1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304201643_8ed35c98-3951-44b7-a468-6f4fe501fb87
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>
Job running in-process (local Hadoop)
2023-03-04 20:16:47,062 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local529990917_0006
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>
Job running in-process (local Hadoop)
2023-03-04 20:16:48,770 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local406906063_0007
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1109244 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1109244 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
temp.productline      temp.qo
Classic Cars      33992
Time taken: 5.006 seconds, Fetched: 1 row(s)
hive>

```

### 3-Calculate the total sales for each quarter

```

hive> select qtr_id,sum(sales)
      > from vehicle_sales_order_orc
      > group by qtr_id;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304202024_d4d598a0-3508-49f7-941f-f29e92695b61
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>
Job running in-process (local Hadoop)
2023-03-04 20:20:26,601 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local459597530_0008
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1153744 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
qtr_id  _c1
1      2350510
2      2047855
3      1758673
4      3874271
Time taken: 2.261 seconds, Fetched: 4 row(s)
hive>

```

#### 4-In which quarter sales was minimum

```

hive> with temp as (
  > select qtr_id,sum(sales) as sales
  > from vehicle_sales_order_orc
  > group by qtr_id
  > )
  > select * from temp
  > order by sales asc
  > limit 1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304202234_8a367679-c1a9-4e88-aea1-85f1401f11ba
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>
Job running in-process (local Hadoop)
2023-03-04 20:22:36,700 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1446558930_0009
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>
Job running in-process (local Hadoop)
2023-03-04 20:22:38,394 Stage-2 map = 100%,  reduce = 100%
Ended Job = job_local1549131299_0010
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1198244 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1198244 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
temp.qtr_id      temp.sales
3              1758673
Time taken: 4.088 seconds, Fetched: 1 row(s)
hive>

```

**5-In which country sales was maximum and in which country sales was minimum**

**Country with minimum sales**

```

hive> with temp as (
  > select country,sum(sales) as sales
  > from vehicle_sales_order_orc
  > group by country)
  > select * from temp order by sales asc limit 1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304203613_82ff5672-45d1-4390-b7ec-5d166a5a2be9

```

```

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>
Job running in-process (local Hadoop)
2023-03-04 20:36:15,760 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local1752526632_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.reduces=<number>
Job running in-process (local Hadoop)
2023-03-04 20:36:17,517 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local453174816_0016
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1337090 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1337090 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
temp.country    temp.sales
Ireland 57749
Time taken: 4.203 seconds, Fetched: 1 row(s)
hive>

```

### Country with maximum sales

```

hive> with temp as (
  > select country,sum(sales) as sales
  > from vehicle_sales_order_orc
  > group by country)
  > select * from temp order by sales desc limit 1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the

```

```

future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304203753_82bcd9b9-7a80-4436-ab5c-6fd6413c4053
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>
Job running in-process (local Hadoop)
2023-03-04 20:37:57,276 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local2119296277_0017
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>
Job running in-process (local Hadoop)
2023-03-04 20:37:59,065 Stage-2 map = 100%,  reduce = 100%
Ended Job = job_local632059477_0018
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1383372 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1383372 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
temp.country      temp.sales
USA              3627511
Time taken: 5.359 seconds, Fetched: 1 row(s)
hive>

```

## 6-Calculate quarterly sales for each city

```

hive> select qtr_id,country,sum(sales)
> from vehicle_sales_order_orc

```



```

> group by qtr_id,country
> order by country,qtr_id;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304204132_f286d683-79f1-4323-a104-97b3b3961713
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>
Job running in-process (local Hadoop)
2023-03-04 20:41:34,874 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local11345370608_0020
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>
Job running in-process (local Hadoop)
2023-03-04 20:41:36,725 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local11774137273_0021
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1477828 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1477828 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
qtr_id  country _c2
1       Australia      136362
2       Australia      118467
3       Australia      139847
4       Australia      235859
1       Austria 8773
2       Austria 98091
3       Austria 6692
4       Austria 88480

```

1	Belgium	35424
2	Belgium	10121
3	Belgium	49390
4	Belgium	13462
2	Canada	89545
3	Canada	43326
4	Canada	91173
1	Denmark	58866
2	Denmark	62084
4	Denmark	124658
1	Finland	126832
2	Finland	48827
3	Finland	111801
4	Finland	42078
1	France	327829
2	France	292416
3	France	121489
4	France	369036
1	Germany	48694
3	Germany	34987
4	Germany	136760
1	Ireland	38781
3	Ireland	18968
1	Italy	56172
2	Italy	41503
3	Italy	150522
4	Italy	126427
1	Japan	88669
2	Japan	43593
4	Japan	55882
1	Norway	54694
3	Norway	50501
4	Norway	202230
1	Philippines	55239
4	Philippines	38767
1	Singapore	28391
2	Singapore	92018
3	Singapore	90240
4	Singapore	77802
1	Spain	357618
2	Spain	343771
3	Spain	69704
4	Spain	444437

```

1      Sweden  41351
3      Sweden  53933
4      Sweden  114702
1      Switzerland  50427
3      Switzerland  67273
1      UK      86391
2      UK      123572
4      UK      268851
1      USA     749997
2      USA     683847
3      USA     750000
4      USA     1443667
Time taken: 4.426 seconds, Fetched: 63 row(s)
hive>

```

## 7-Find a month for each year in which maximum number of quantities were sold

```

hive> with temp as (select month_id,year_id,sum(sales) as sales from
vehicle_sales_order_orc group by month_id,year_id)
> ,temp1 as (select *,row_number() over (partition by year_id order by
sales desc) as flag from temp)
> select * from temp1 where flag=1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the
future versions. Consider using a different execution engine (i.e. spark,
tez) or using Hive 1.X releases.
Query ID = root_20230304205051_9ad54bc0-7e5e-41b9-a4c4-4f54898dfd11
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>
Job running in-process (local Hadoop)
2023-03-04 20:50:53,078 Stage-1 map = 100%, reduce = 100%

```

```
Ended Job = job_local1190994864_0027
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>
Job running in-process (local Hadoop)
2023-03-04 20:50:55,233 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local1225640861_0028
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 1659812 HDFS Write: 59938 SUCCESS
Stage-Stage-2:  HDFS Read: 1659812 HDFS Write: 59938 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
temp1.month_id  temp1.year_id  temp1.sales    temp1.flag
11             2003          1029698 1
11             2004          1088908 1
5              2005          457808 1
Time taken: 4.245 seconds, Fetched: 3 row(s)
hive>
```