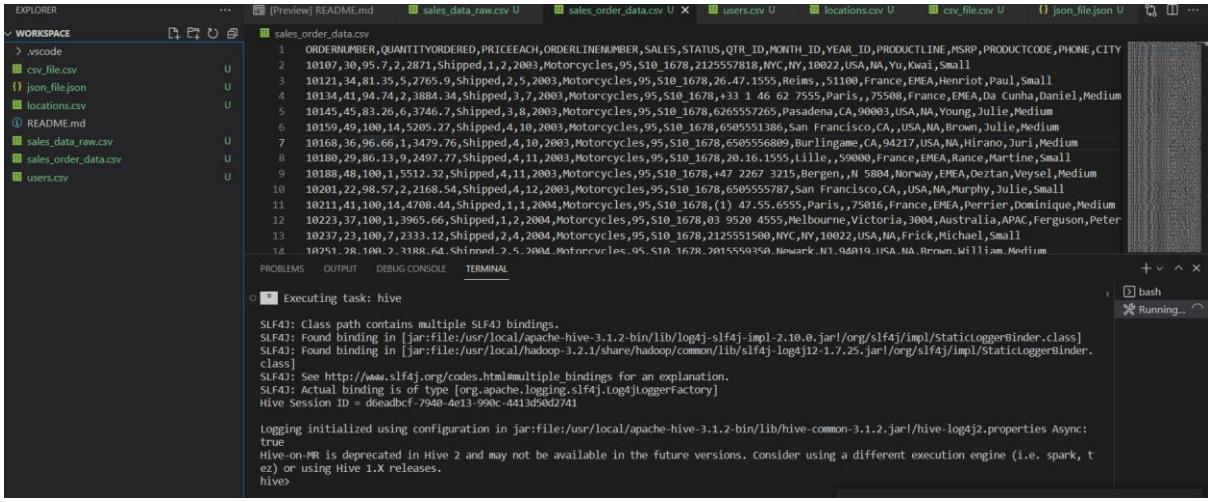
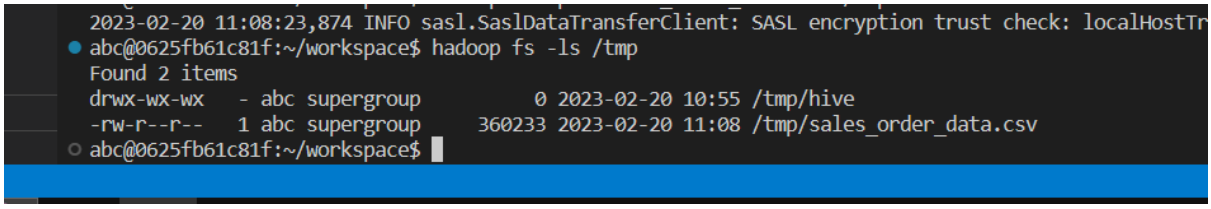
# Hive-Assignment-1

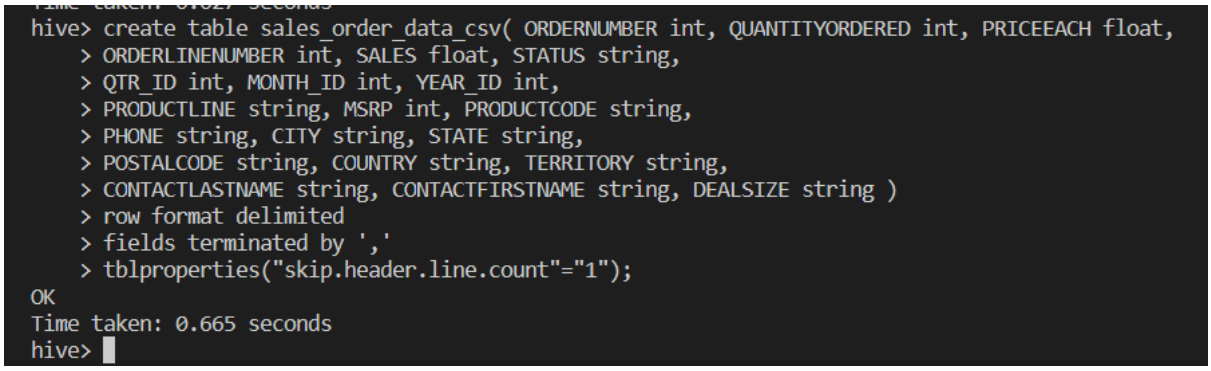
## 1. Download vechile sales data -> https://github.com/shashank-mishra219/Hive-Class/blob/main/sales_order_data.csv




## 2. Store raw data into hdfs location




## 3. 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



## 4. Load data from hdfs path into "sales_order_csv"

```
hive> load data inpath '/tmp/sales_order_data.csv' into table sales_order_data_csv;
Loading data to table hivedb.sales_order_data_csv
OK
Time taken: 1.032 seconds
```

## 5. Create an internal hive table which will store data in ORC format "sales_order_orc"

```
hive> create table sales_order_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: 0.101 seconds
```

## 6. Load data from "sales_order_csv" into "sales_order_orc"

```
hive> from sales_order_data_csv insert overwrite table sales_order_data_orc select *;
Query ID = abc_20230220114716_249025da-bf39-4c9a-a6dd-8837517122a6
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_1676872878406_0001, Tracking URL = http://700013528be7:8088/proxy/application_1676872878406_0001/
Kill Command = /usr/local/hadoop/bin/mapred job  -kill job_1676872878406_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2023-02-20 11:47:30,347 Stage-1 map = 0%,  reduce = 0%
2023-02-20 11:47:38,607 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 13.42 sec
2023-02-20 11:47:43,751 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 15.83 sec
MapReduce Total cumulative CPU time: 15 seconds 830 msec
Ended Job = job_1676872878406_0001
Stage-4 is selected by condition resolver.
```

# Perform below menioned queries on "sales_order_orc" table :

## a. Calculate total sales per year

```
hive> from sales_order_data_csv insert overwrite table sales_order_data_orc select *;
Query ID = abc_20230220114716_249025da-bf39-4c9a-a6dd-8837517122a6
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_1676872878406_0001, Tracking URL = http://700013528be7:8088/proxy/application_1676872878406_0001/
Kill Command = /usr/local/hadoop/bin/mapred job  -kill job_1676872878406_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2023-02-20 11:47:30,347 Stage-1 map = 0%,  reduce = 0%
2023-02-20 11:47:38,607 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 13.42 sec
2023-02-20 11:47:43,751 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 15.83 sec
MapReduce Total cumulative CPU time: 15 seconds 830 msec
Ended Job = job_1676872878406_0001
Stage-4 is selected by condition resolver.
```

```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.69 sec   HDFS Read: 44239 HDFS Write: 193 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 690 msec
OK
year_id total_sales
2003    3516979.547241211
2004    4724162.593383789
2005    1791486.7086791992
Time taken: 24.568 seconds, Fetched: 3 row(s)
```

## b. Find a product for which maximum orders were placed

```
hive> select productline,max(quantityordered) as max_ordered from sales_order_data_csv
    > group by (productline)
    > order by (max_ordered) desc
    > limit 1 ;
Query ID = abc_20230220122233_b83541dd-ad78-486a-8233-b30a62201eec
Total jobs = 2
Launching Job 1 out of 2
```

```
productline     max_ordered
Classic Cars    97
Time taken: 231.581 seconds, Fetched: 1 row(s)
```

## c. Calculate the total sales for each quarter

```
hive> select qtr_id,sum(sales) as each_quarter from sales_order_data_csv
    > group by(qtr_id);
Query ID = abc_20230220133920_4ebf0388-b3a5-4ec1-9bd1-4cd47fe6df90
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
```

```
qtr_id  each_quarter
1       2350817.726501465
2       2048120.3029174805
3       1758910.808959961
4       3874780.010925293
Time taken: 23.829 seconds, Fetched: 4 row(s)
```

## d. In which quarter sales was minimum

```
hive> select qtr_id,sum(sales) as minimum_sales from sales_order_data_csv
    > group by(qtr_id)
    > order by (minimum_sales) asc
    > limit 1;
Query ID = abc_20230220134329_c33f1422-06b3-4cff-b8b0-576cd42a7b40
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>
```

```
qtr_id  minimum_sales
3       1758910.808959961
Time taken: 55.008 seconds, Fetched: 1 row(s)
```

## e. In which country sales was maximum and in which country sales was minimum

```
hive> select country,max(sales) as maximum_sales
    > from sales_order_data_csv
    > group by country
    > order by maximum_sales desc
    > limit 1;
```

```
country maximum_sales
USA     14082.8
Time taken: 49.579 seconds, Fetched: 1 row(s)
```

```
hive> select country,min(sales) as minimum_sales
    > from sales_order_data_csv
    > group by country
    > order by minimum_sales asc
    > limit 1;
Query ID = abc_20230220151654_64e2cc49-34d8-427e-9d4f-ab2b1500712a
```

```
OK
country minimum_sales
France  482.13
Time taken: 230.643 seconds, Fetche
```

## f. Calculate quartelry sales for each city

```
Time taken: 51.268 seconds, Fetched: 300 row(s)
hive> select country,sum(sales),qtr_id from sales_order_data_orc
    > group by country,qtr_id
    > order by qtr_id;
Query ID = abc_20230220191130_71cf5e23-c7d3-4927-83ab-0b228aa0a180
Total jobs = 2
OK
Australia       136380.02014160156      1
USA     750090.1377563477       1
Switzerland     50432.549560546875      1
UK      86401.82000732422       1
Austria 8775.159912109375       1
Belgium 35428.24987792969       1
Denmark 58871.110107421875      1
Finland 126851.71044921875      1
France  327873.8992919922       1
Germany 48698.82922363281       1
Ireland 38784.470458984375      1
Italy   56181.320068359375      1
Japan   88682.02990722656       1
Norway  54701.999755859375      1
Philippines     55245.02014160156       1
Singapore       28395.18994140625       1
Spain   357668.4899291992       1
Sweden  41355.719970703125      1
Japan   43597.130432128906      2
Italy   41509.94006347656       2
```

```
Canada  89560.00036621094       2
Belgium 10123.209838867188      2
Austria 98104.24005126953       2
Australia       118485.36022949219      2
USA     683935.8491821289       2
UK      123587.35009765625      2
Spain   343807.2515258789       2
Singapore       92033.77014160156       2
Australia       139861.26989746094      3
USA     750104.1696777344       3
Switzerland     67281.00903320312       3
Sweden  53941.68981933594       3
Spain   69714.09008789062       3
Singapore       90250.07995605469       3
Norway  50508.57019042969       3
Italy   150538.9102783203       3
Ireland 18971.959838867188      3
Germany 34993.92004394531       3
France  121506.76062011719      3
Finland 111815.06994628906      3
Canada  43332.349609375 3
```

```
Belgium 49397.6796875   3
Austria 6693.2802734375 3
Australia       235896.44848632812      4
Austria 88489.85009765625       4
Belgium 13463.480224609375      4
Canada  91186.2099609375        4
Denmark 124674.15991210938      4
Finland 42083.499755859375      4
France  369083.17169189453      4
Germany 136779.34045410156      4
Italy   126444.14056396484      4
Japan   55888.65026855469       4
Norway  202253.1297607422       4
Philippines     38770.71032714844       4
Singapore       77809.37023925781       4
Spain   444497.0908203125       4
Sweden  114716.80041503906      4
UK      268891.2888183594       4
USA     1443852.669128418       4
Time taken: 51.405 seconds, Fetched: 63 row(s)
```

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

```
hive> SELECT t.YEAR_ID, t.MONTH_ID, t.MAX_QUANTITY
    > FROM (
    >   SELECT YEAR_ID, MONTH_ID, MAX(QUANTITYORDERED) as MAX_QUANTITY
    >   FROM sales_order_data_csv
    >   GROUP BY YEAR_ID, MONTH_ID
    > ) t
    > JOIN (
    >   SELECT YEAR_ID, MAX(MAX_QUANTITY) as MAX_QUANTITY
    >   FROM (
    >     SELECT YEAR_ID, MONTH_ID, MAX(QUANTITYORDERED) as MAX_QUANTITY
    >     FROM sales_order_data_csv
    >     GROUP BY YEAR_ID, MONTH_ID
    >   ) s
    >   GROUP BY YEAR_ID
    > ) s
    > ON t.YEAR_ID = s.YEAR_ID AND t.MAX_QUANTITY = s.MAX_QUANTITY
    > ORDER BY t.YEAR_ID
    > ;
```

```
OK
2003    11      50
2003    10      50
2003    9       50
2003    6       50
2003    5       50
2003    4       50
2003    3       50
2003    2       50
2003    1       50
2004    11      55
2005    4       97
Time taken: 109.513 seconds, Fetched: 11 row(s)
```