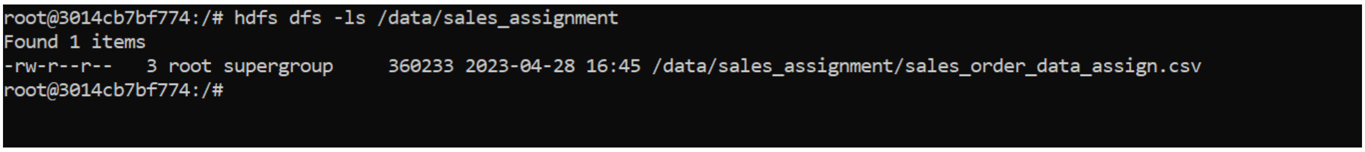
1. Download vechile sales data -> https://github.com/shashank-mishra219/Hive-Class/blob/main/sales_order_data.csv

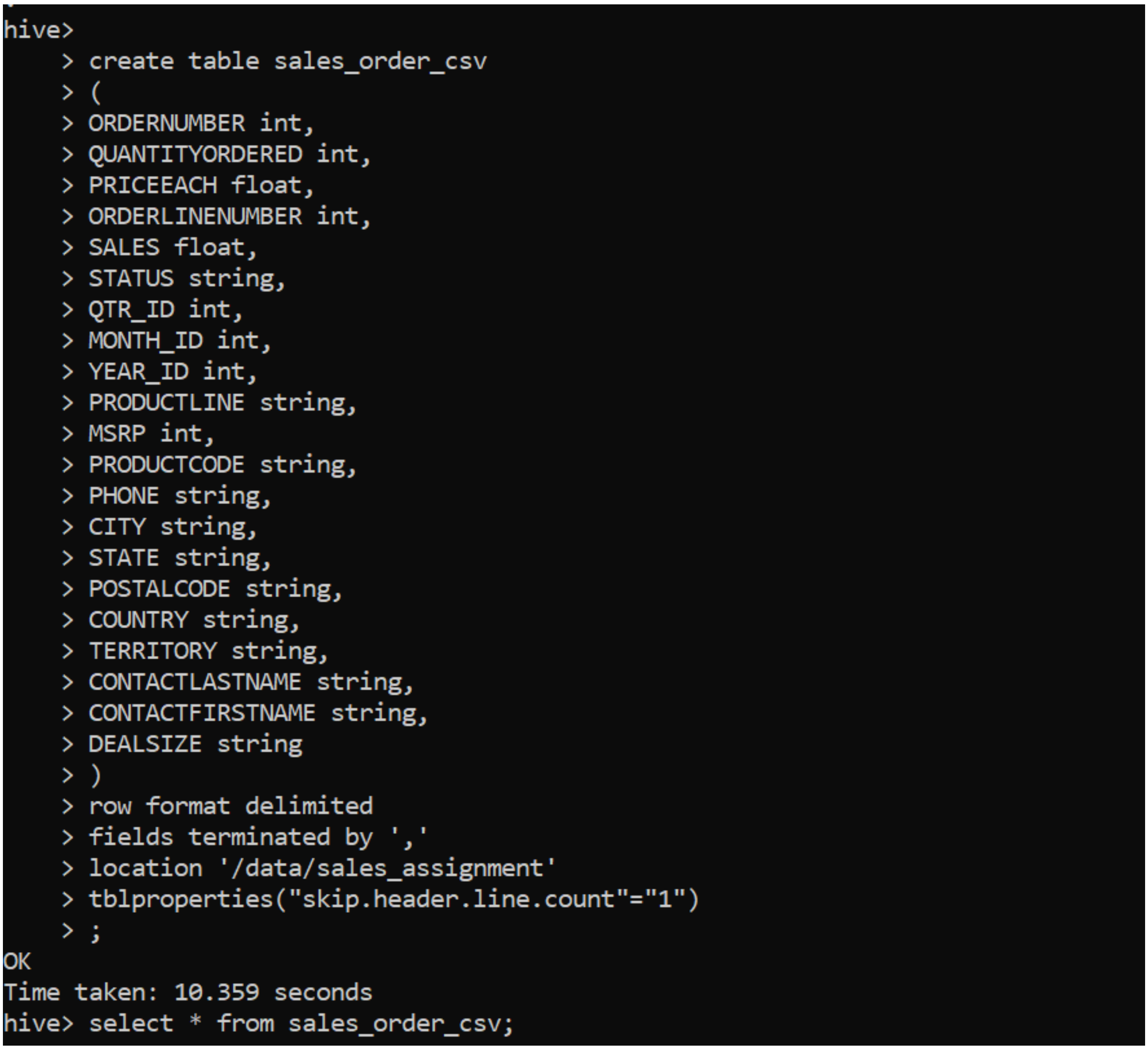
-- **git clone 'https://github.com/shashank-mishra219/Hive-Class.git'**

2. Store raw data into hdfs location



```
root@3014cb7bf774:/# hdfs dfs -ls /data/sales_assignment
Found 1 items
-rw-r--r--   3 root supergroup     360233 2023-04-28 16:45 /data/sales_assignment/sales_order_data_assign.csv
root@3014cb7bf774:/#
```

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

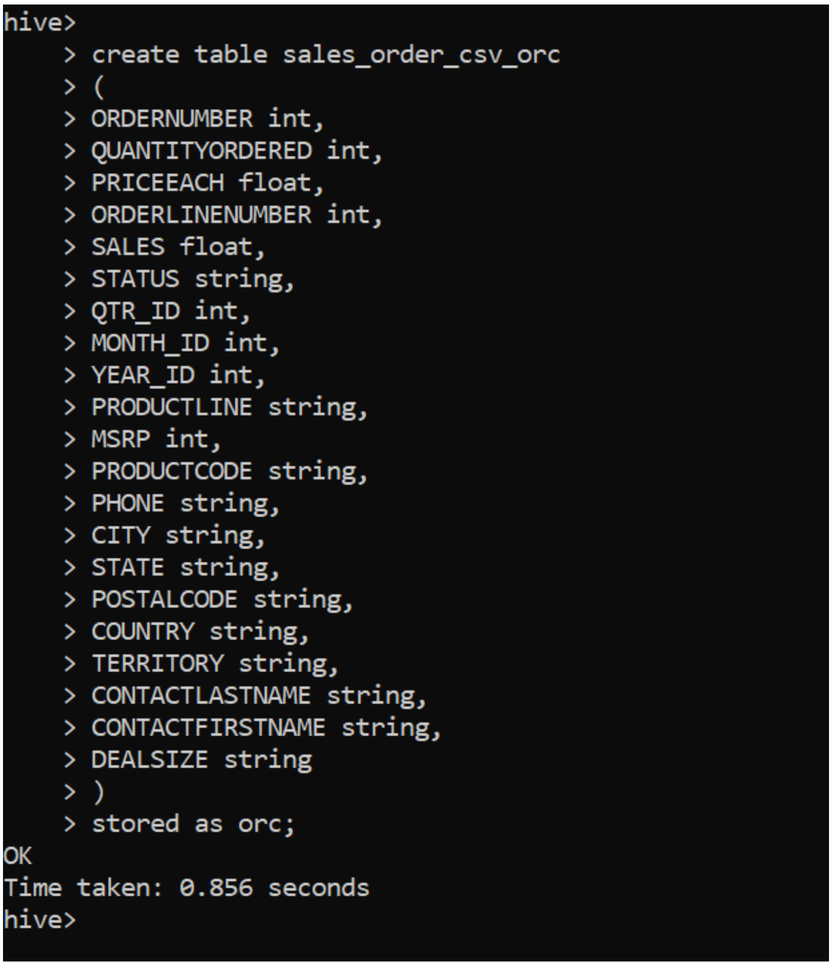


```
hive>
    > create table sales_order_csv
    > (
    > ORDERNUMBER int,
    > QUANTITYORDERED int,
    > PRICEEACH float,
    > ORDERLINENUMBER int,
    > SALES float,
    > STATUS string,
    > QTR_ID int,
    > MONTH_ID int,
    > YEAR_ID int,
    > PRODUCTLINE string,
    > MSRP int,
    > PRODUCTCODE 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 ','
    > location '/data/sales_assignment'
    > tblproperties("skip.header.line.count"="1")
    > ;
OK
Time taken: 10.359 seconds
hive> select * from sales_order_csv;
```

4. Load data from hdfs path into "sales_order_csv"

-- **loaded data while creating the csv table**

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



```
hive>
    > create table sales_order_csv_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: 0.856 seconds
hive>
```

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

-- **from sales_order_csv insert overwrite table sales_order_csv_orc select *;**

Perform below menioned queries on "sales_order_orc" table :

a. Calculatye total sales per year

-- select year_id,sum(sales) as total_sales from sales_order_csv group by year_id;

b. Find a product for which maximum orders were placed

-- select productcode,count(ordernumber) as orders_placed from sales_order_csv_orc group by productcode order by orders_placed desc limit 1;

c. Calculate the total sales for each quarter

```
-- select qtr_id,sum(sales) as total_sales from sales_order_csv_orc group by qtr_id;
```

d. In which quarter sales was minimum

```
-- select qtr_id,sum(sales) as total_sales from sales_order_csv_orc group by qtr_id order by total_sales desc;
```

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

```
-- select country,sum(sales) as total_sales from sales_order_csv_orc group by country order by total_sales desc;
```

f. Calculate quartelry sales for each city

```
-- select city,qtr_id,sum(sales) as total_sales from sales_order_csv_orc group by city,qtr_id order by city,qtr_id;
```

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

```
-- with yearly_data as (select year_id,month_id,sum(quantityordered) as qty_sold from sales_order_csv_orc  group by
              year_id,month_id)
   select year_id,month_id,max(qty_sold) as max_qty_sold from yearly_data group by year_id,month_id;
```