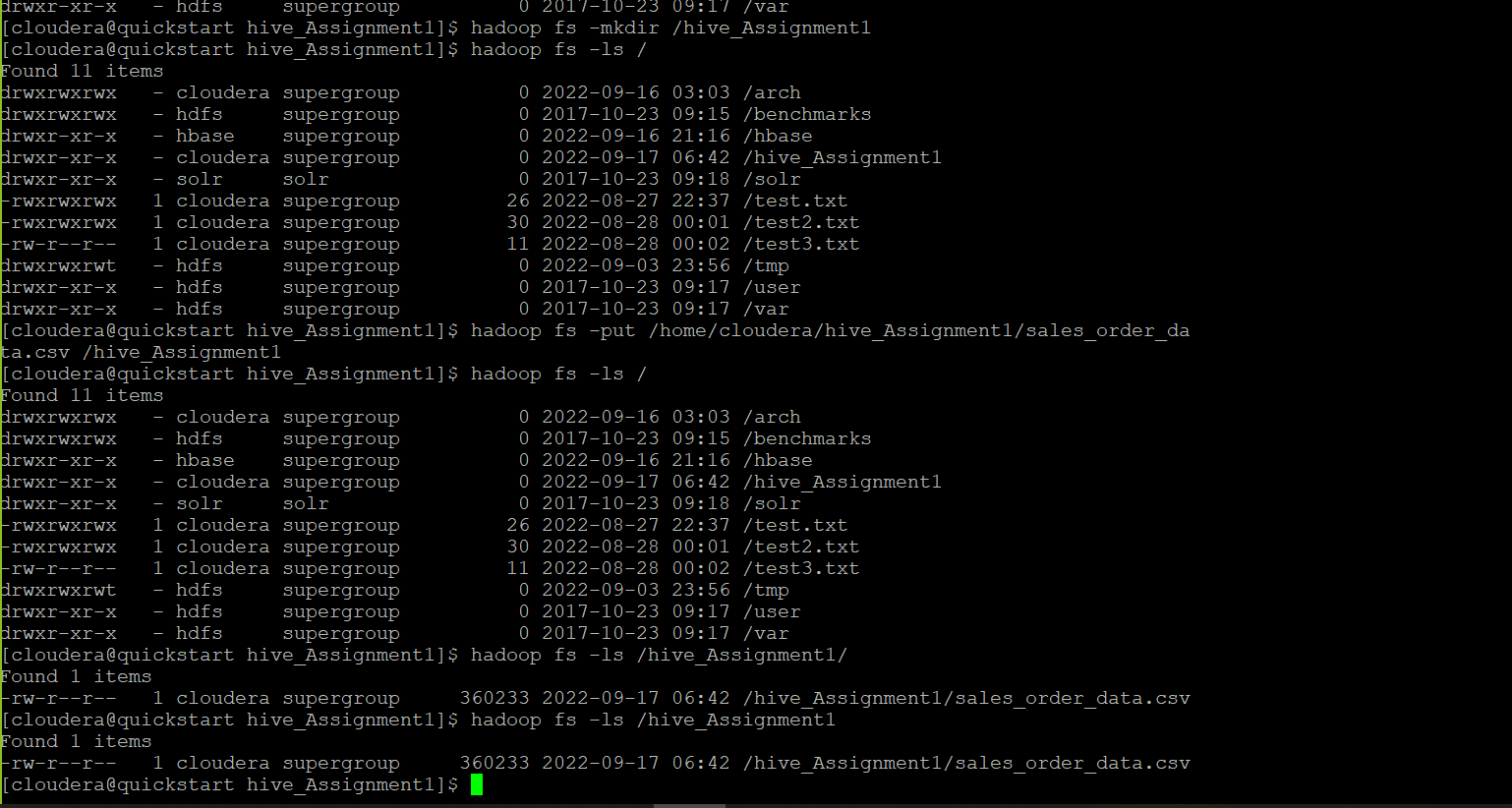
**Task-1 Assignment:**

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

**hadoop fs -mkdir hiveAssignment1**

**hadoop fs -put /home/cloudera/hive\_Assignment1/sales\_order\_da ta.csv /hive\_Assignment1**

**hadoop fs -ls /hive\_Assignment1/**



1. 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 database hive\_Assignment1;**

**hive> show databases;**

**hive> use hive\_assignment1;**

**hive> show tables;**

**hive> create table 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 string,**

**CITY string,**

**STATE string,**

**POSTALCODE int,**

**COUNTRY string,**

**TERRITORY string,**

**CONTACTLASTNAME string,**

**CONTACTFIRSTNAME string,**

**DEALSIZE string**

**)**

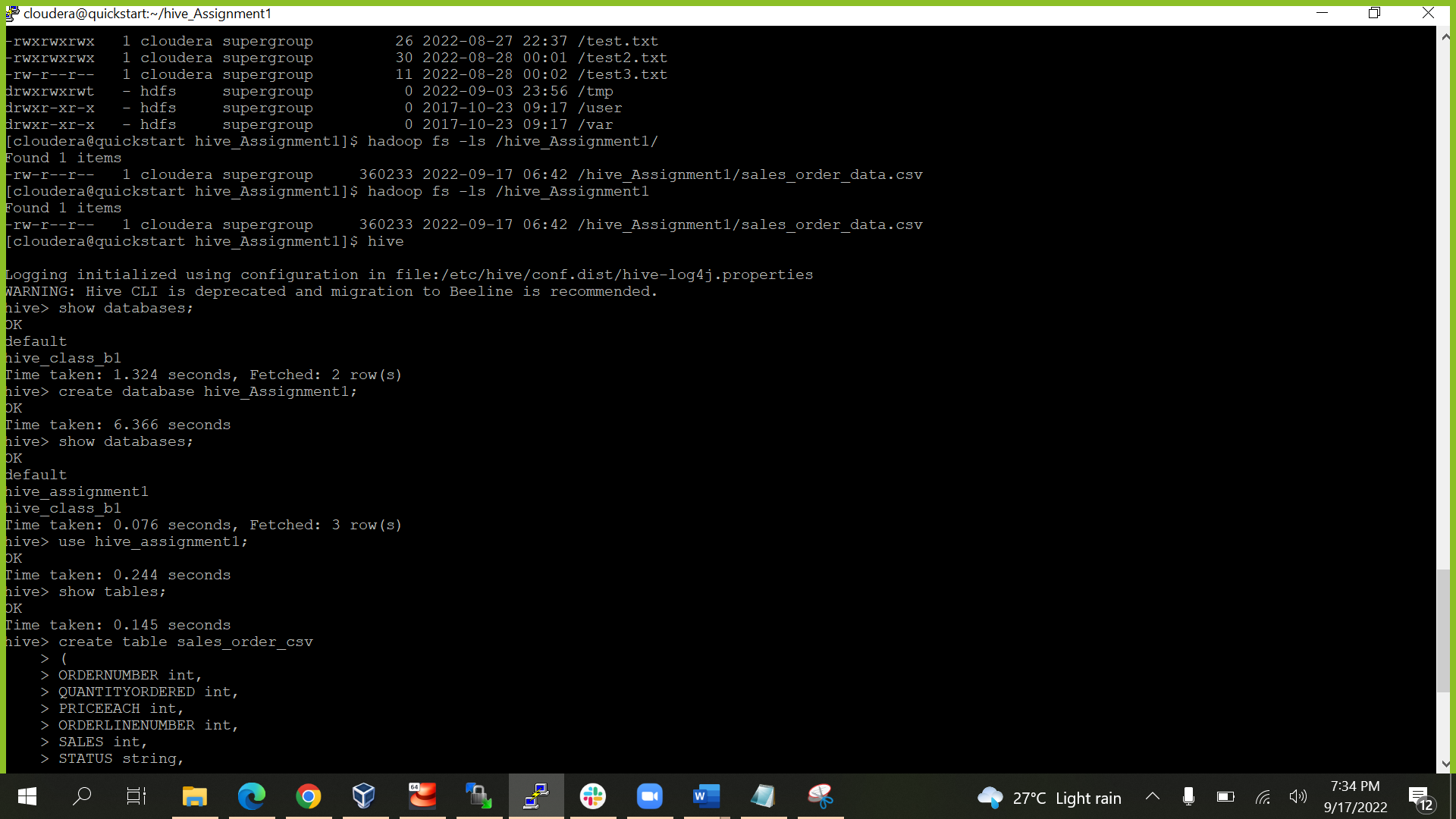
**row format delimited**

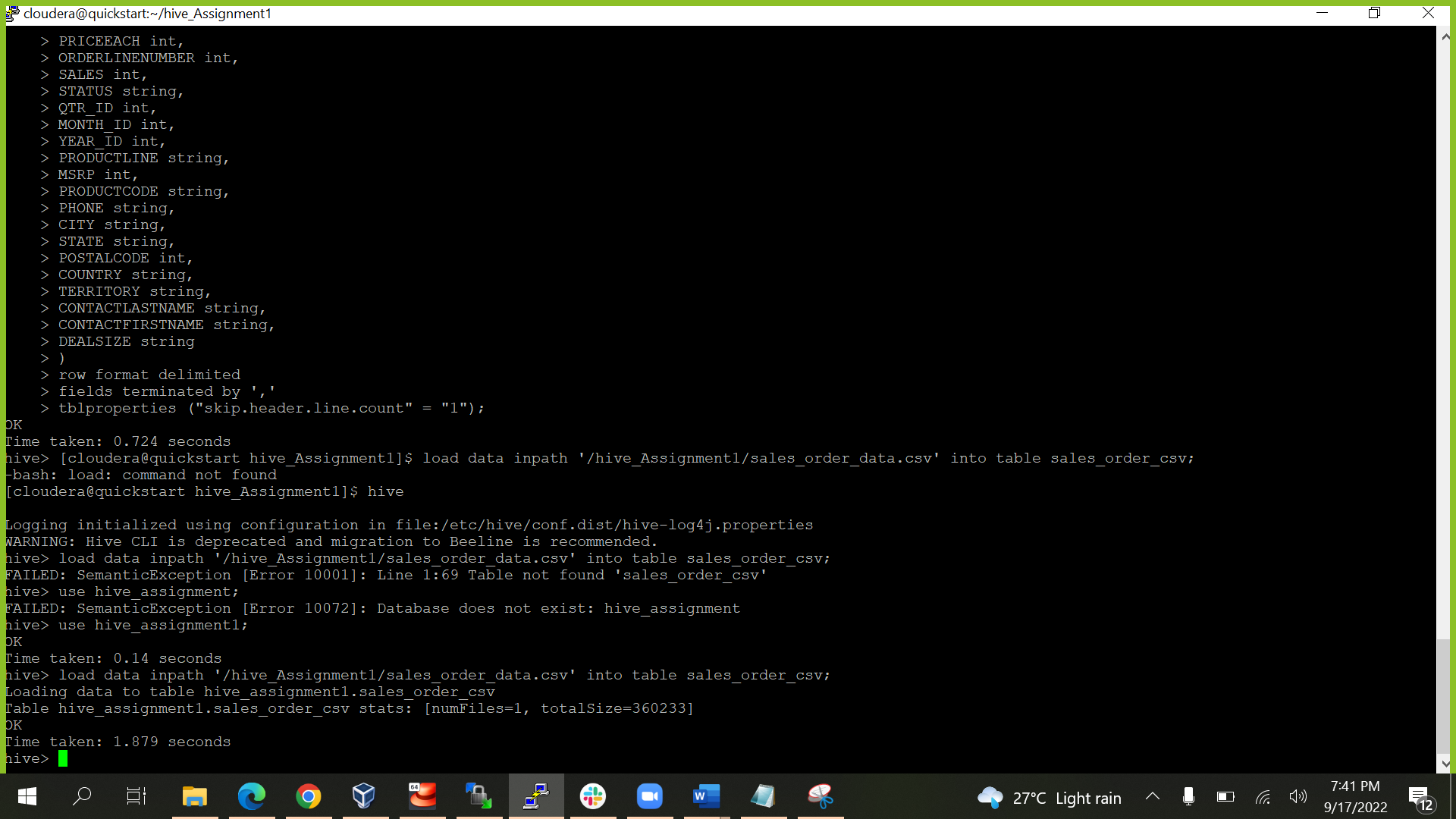
**fields terminated by ','**

**tblproperties ("skip.header.line.count" = "1");**

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

**hive> load data inpath '/hive\_Assignment1/sales\_order\_data.csv' into table sales\_order\_csv;**





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

**create table 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 string,**

**CITY string,**

**STATE string,**

**POSTALCODE int,**

**COUNTRY string,**

**TERRITORY string,**

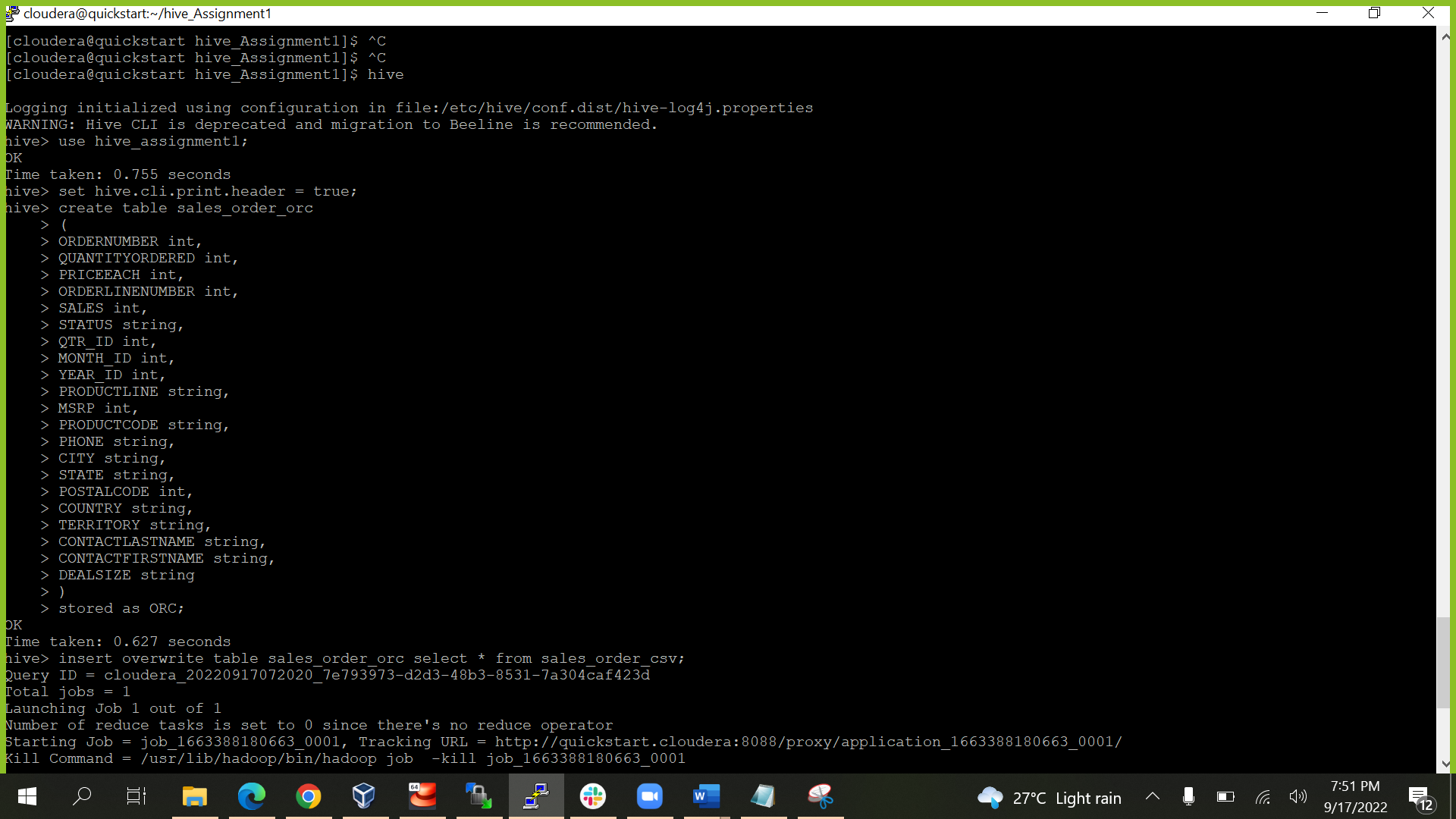
**CONTACTLASTNAME string,**

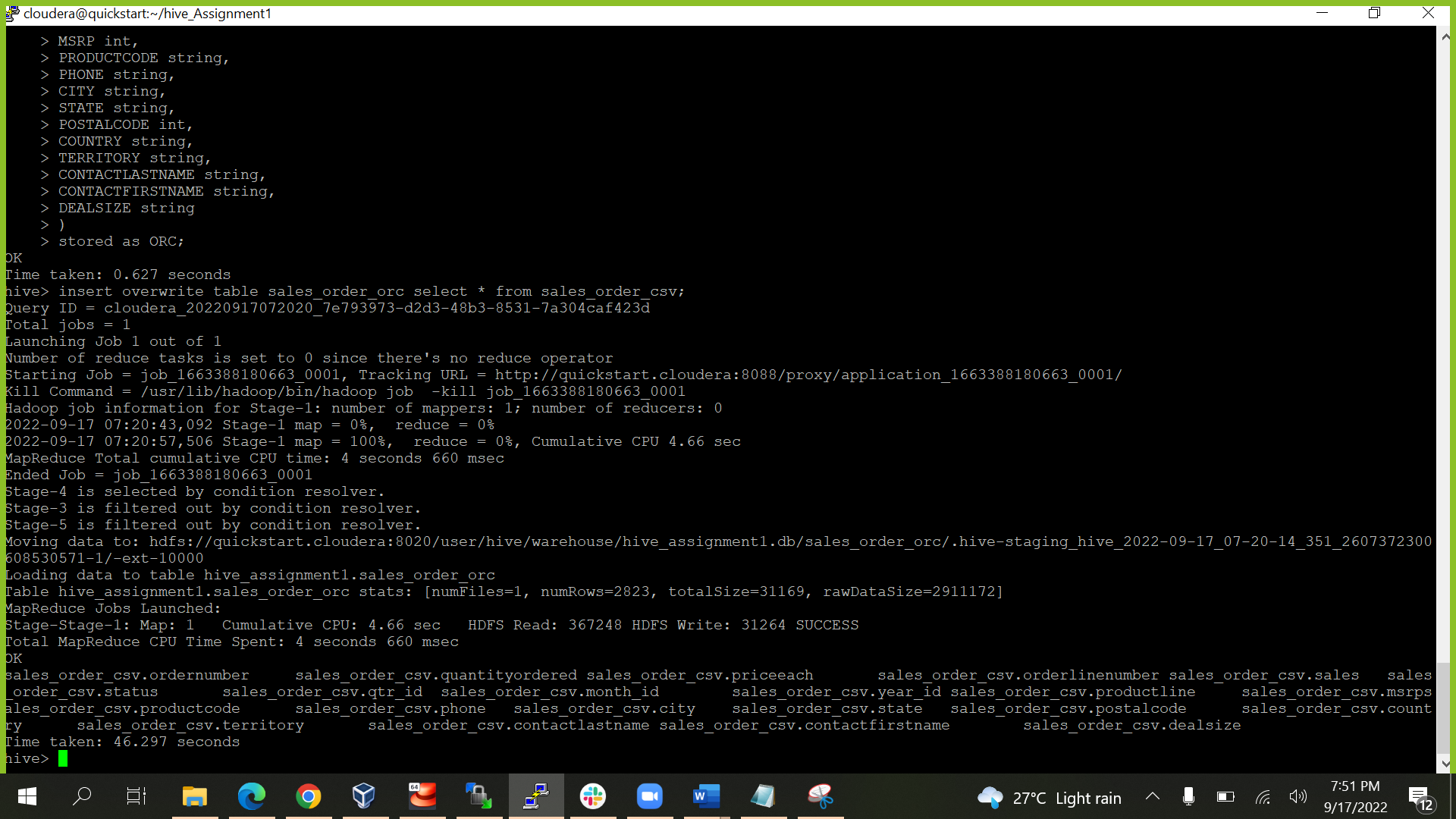
**CONTACTFIRSTNAME string,**

**DEALSIZE string**

**)**

**stored as ORC;**



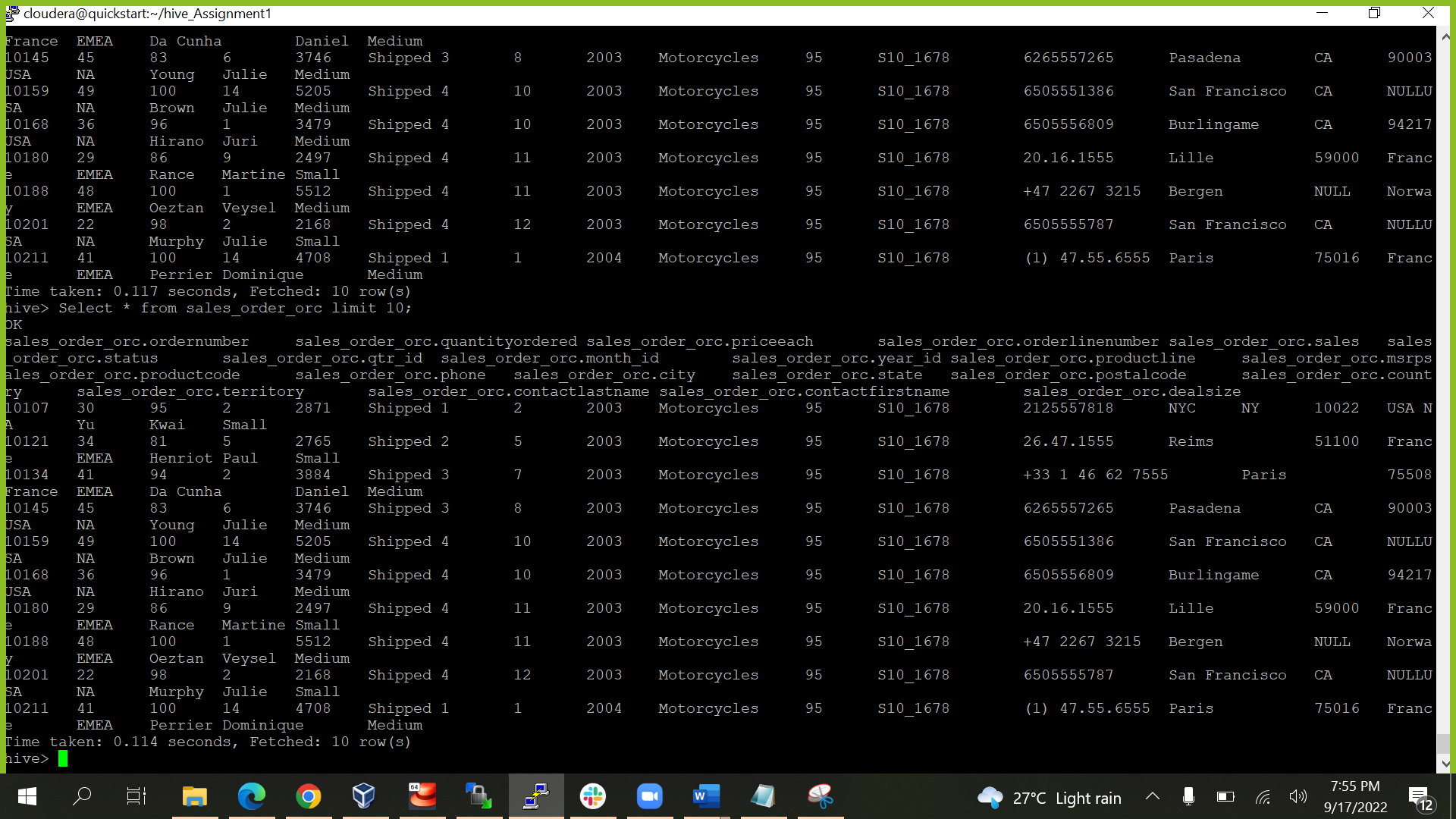


6. Load data from "sales\_order\_csv" into "sales\_order\_orc".

**hive> from sales\_order\_csv insert overwrite table sales\_order\_orc select \*;**

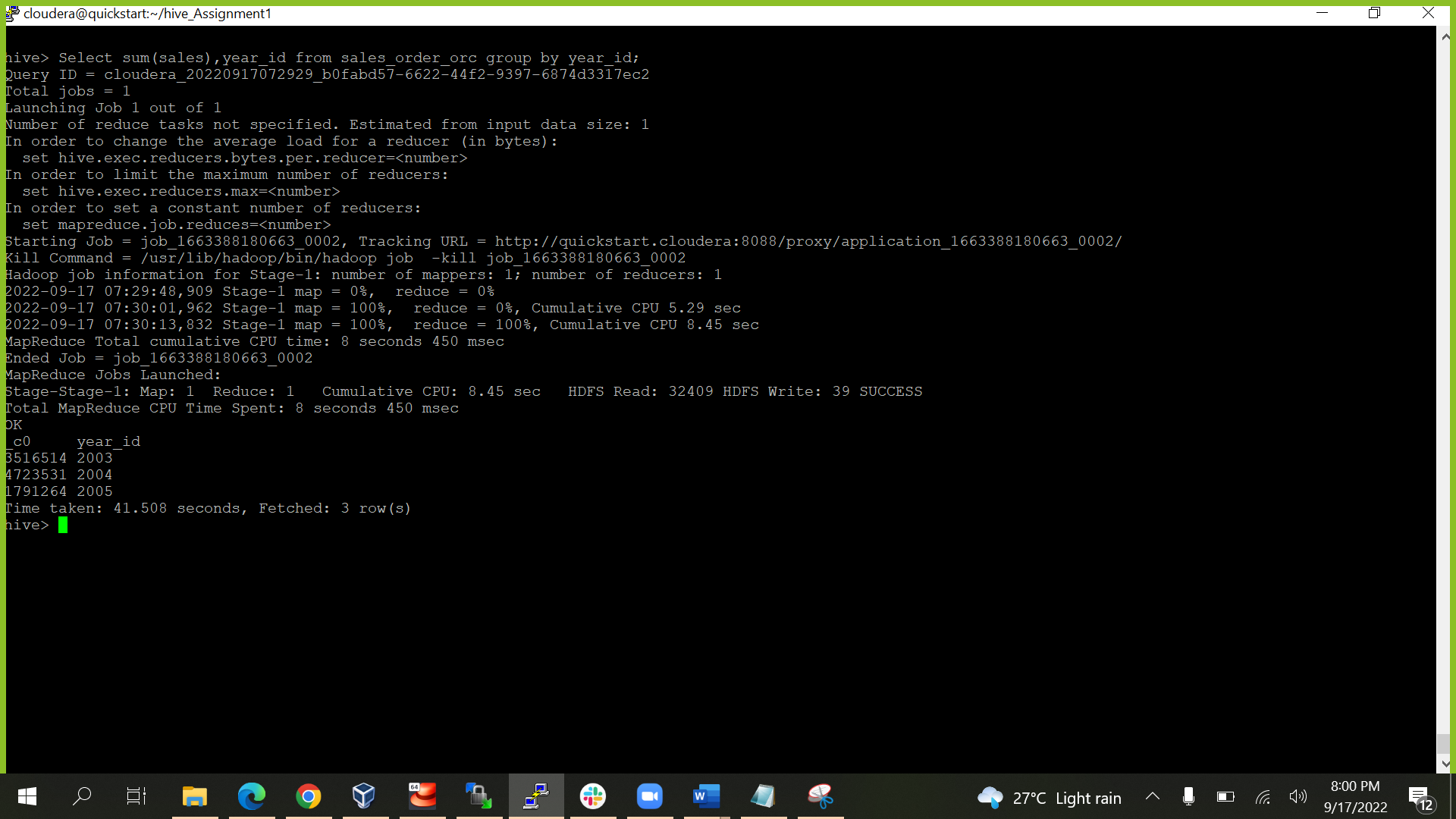
**or**

**hive> insert overwrite table sales\_order\_orc select \* from sales\_order\_csv;**



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

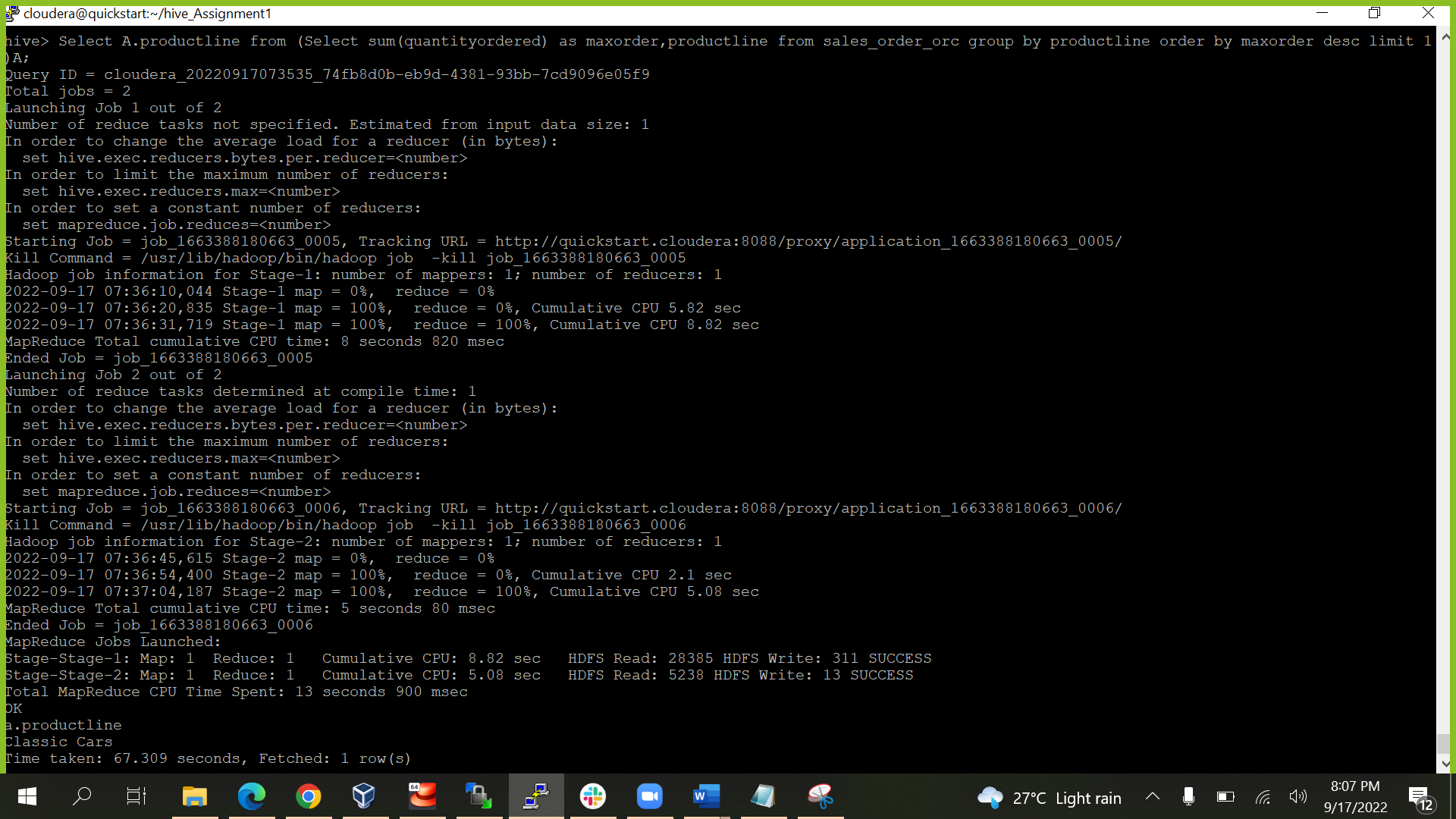
1. Calculate total sales per year.



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

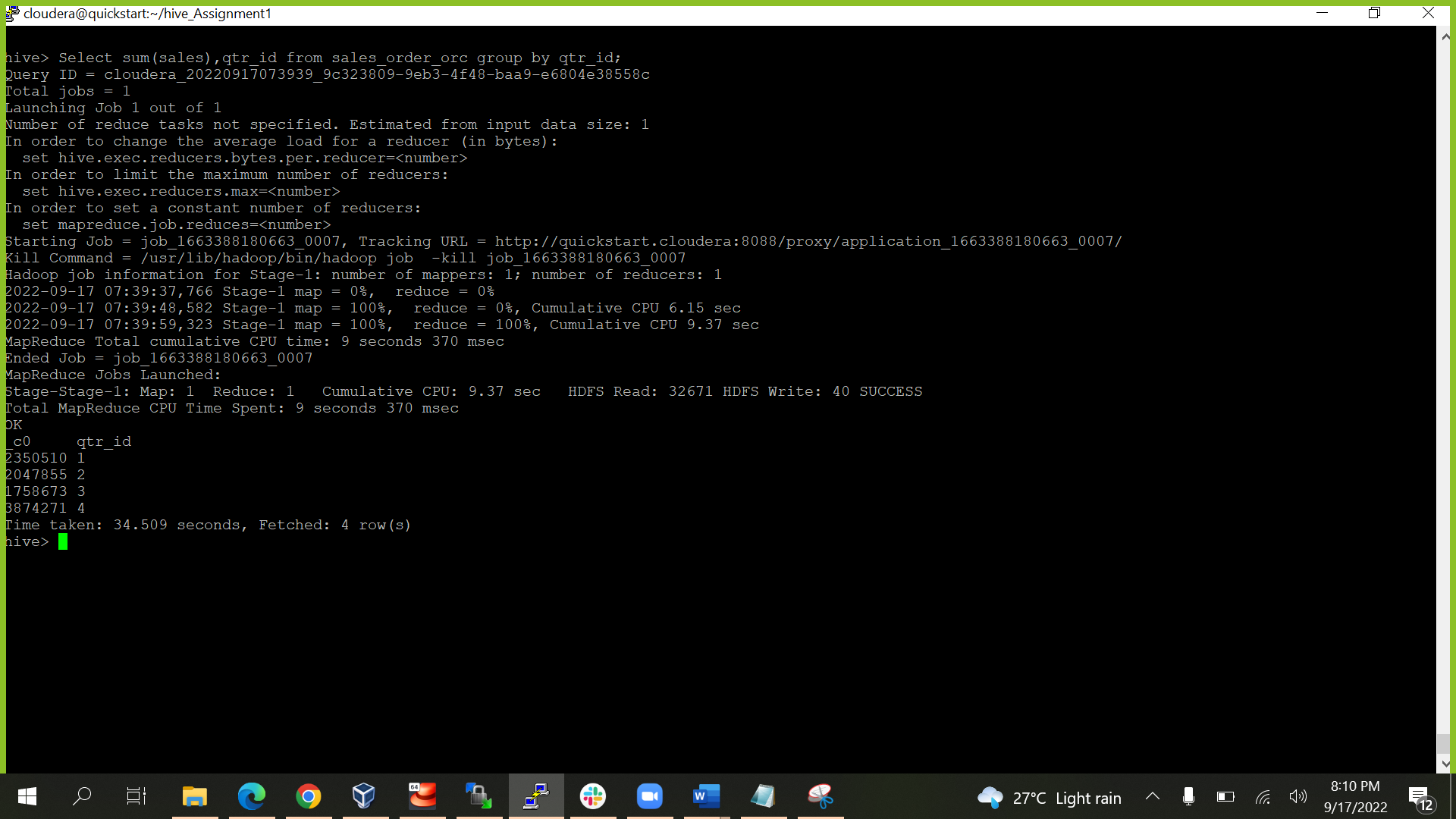
1. Find a product for which maximum orders were placed.

**Select A.productline from (Select sum(quantityordered) as maxorder,productline from sales\_order\_orc group by productline order by maxorder desc limit 1)A;**



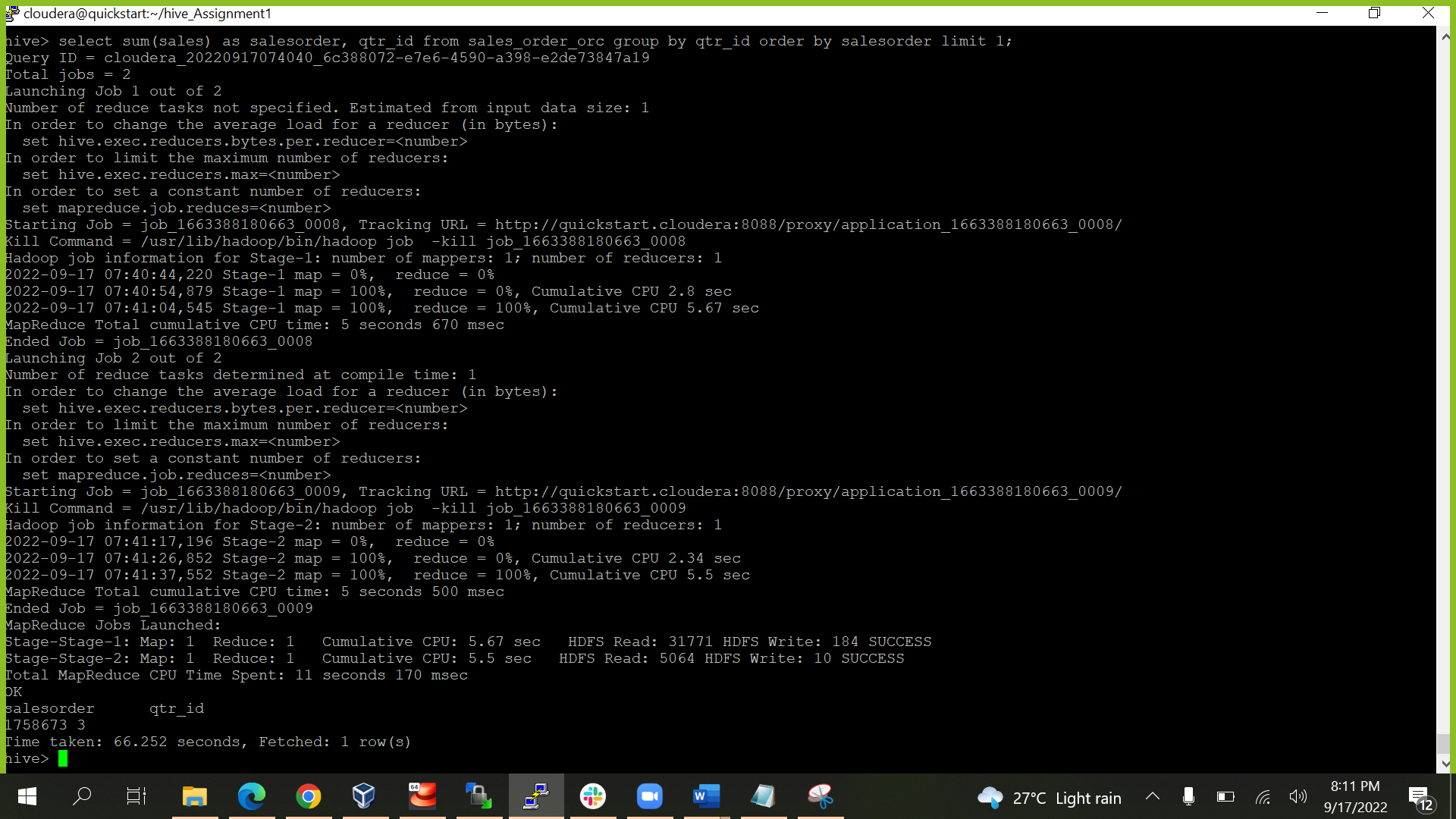
c. Calculate the total sales for each quarter

**Select sum(sales),qtr\_id from sales\_order\_orc group by qtr\_id;**



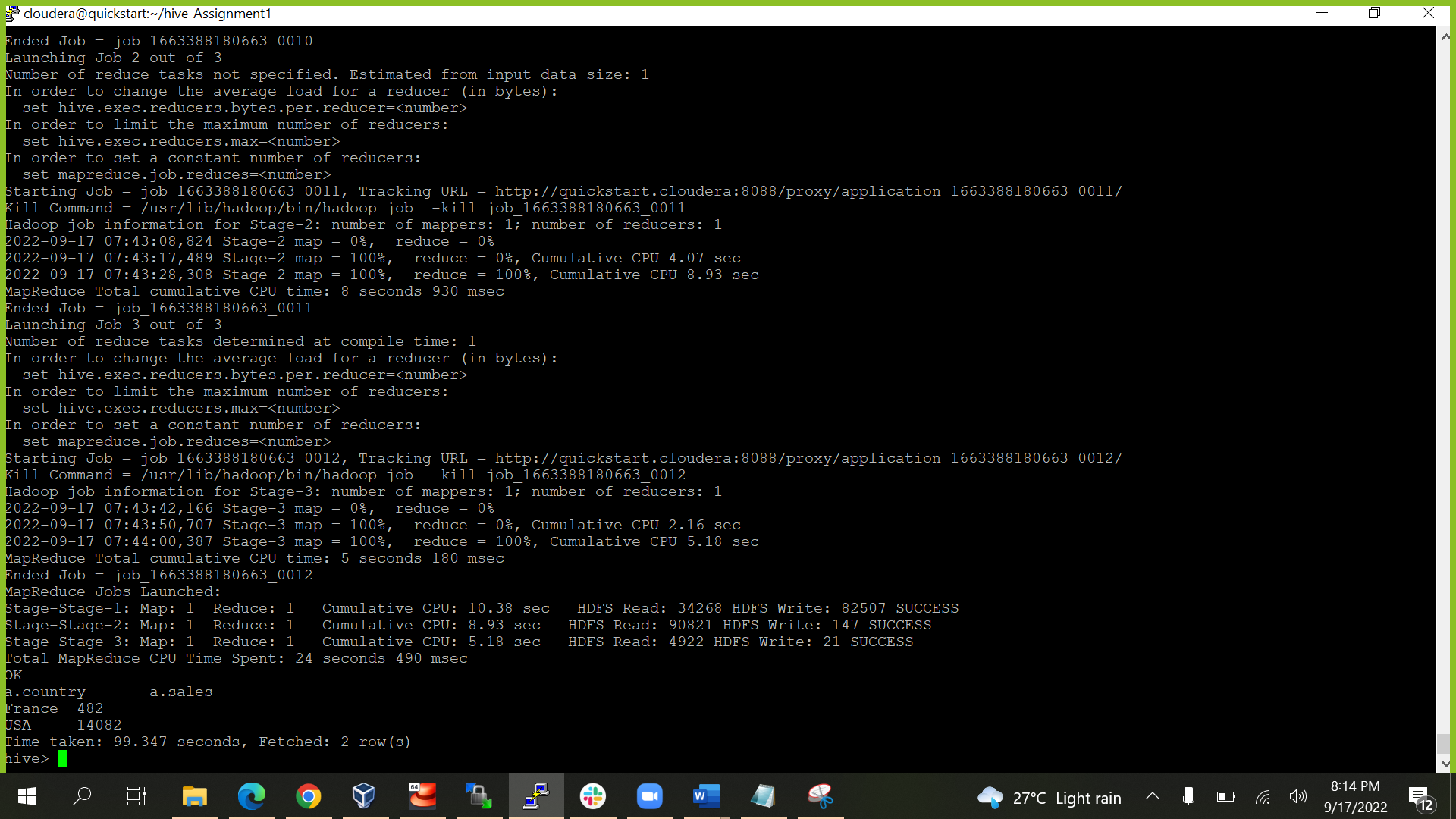
d. In which quarter sales was minimum?

**select sum(sales) as salesorder, qtr\_id from sales\_order\_orc group by qtr\_id order by salesorder limit 1;**



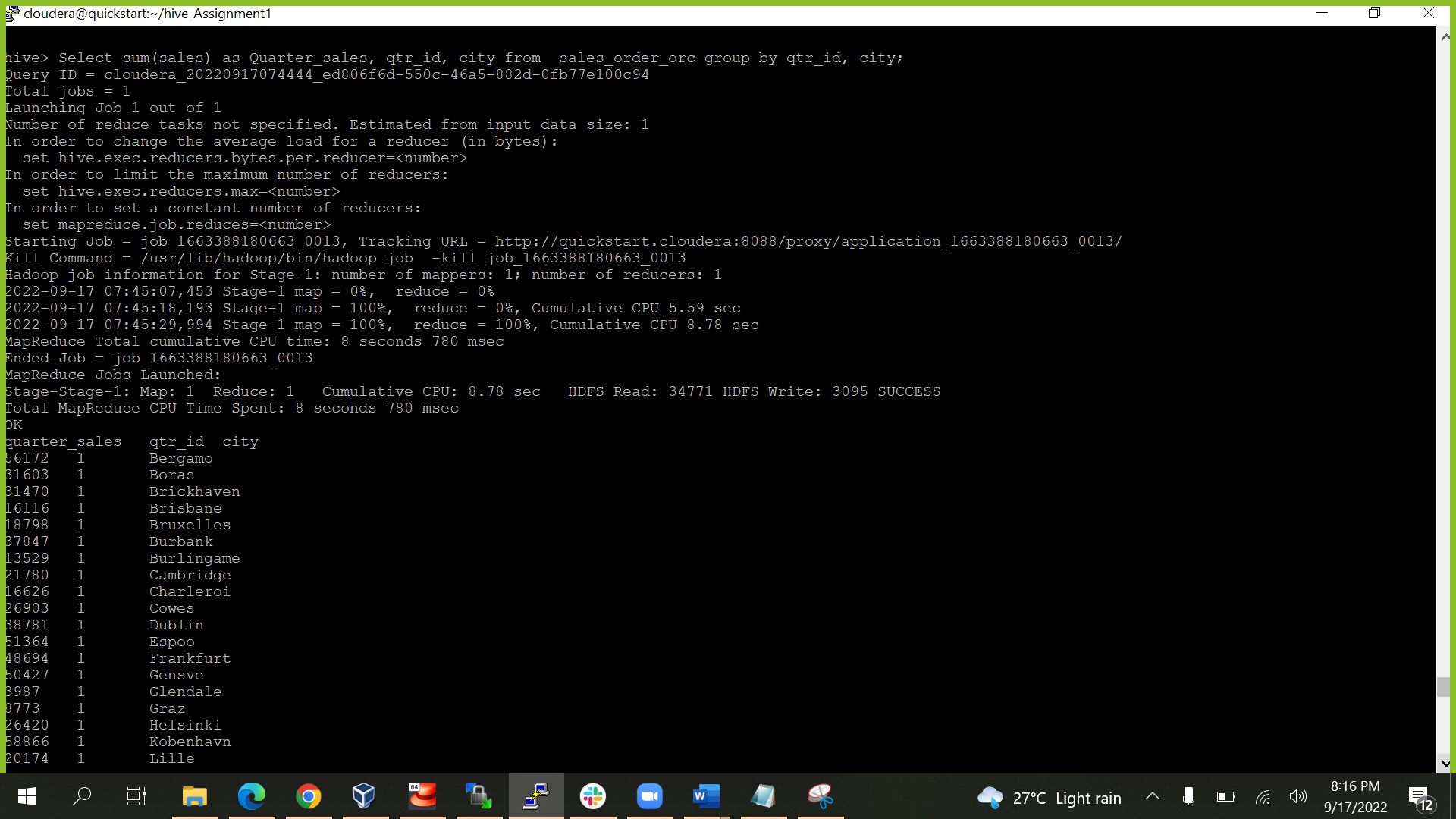
e. In which country sales was maximum and in which country sales was minimum.

**SELECT A.country, A.sales FROM(SELECT country,Sales,RANK() OVER (ORDER BY sales) rnk\_min,RANK() OVER (ORDER BY Sales DESC) rnk\_max FROM sales\_order\_orc)A WHERE rnk\_min = 1 OR rnk\_max = 1 ORDER BY Sales;**



f. Calculate quartelry sales for each city.

**Select sum(sales) as Quarter\_sales, qtr\_id, city from sales\_order\_orc group by qtr\_id, city;**



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

**select month\_id,year\_id,QUANTITYORDERED from (select month\_id,year\_id,QUANTITYORDERED ,dense\_rank() over(partition by year\_id order by QUANTITYORDERED desc) as rnk from sales\_order\_csv)a where a.rnk =1;**

