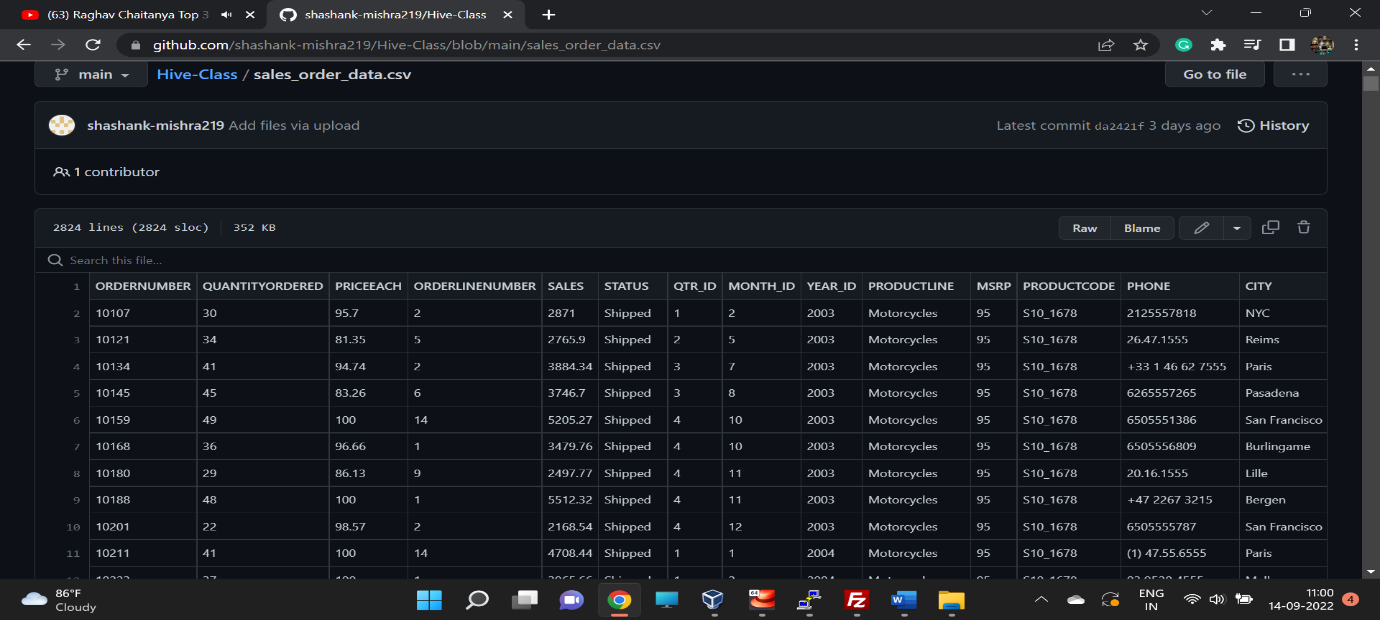
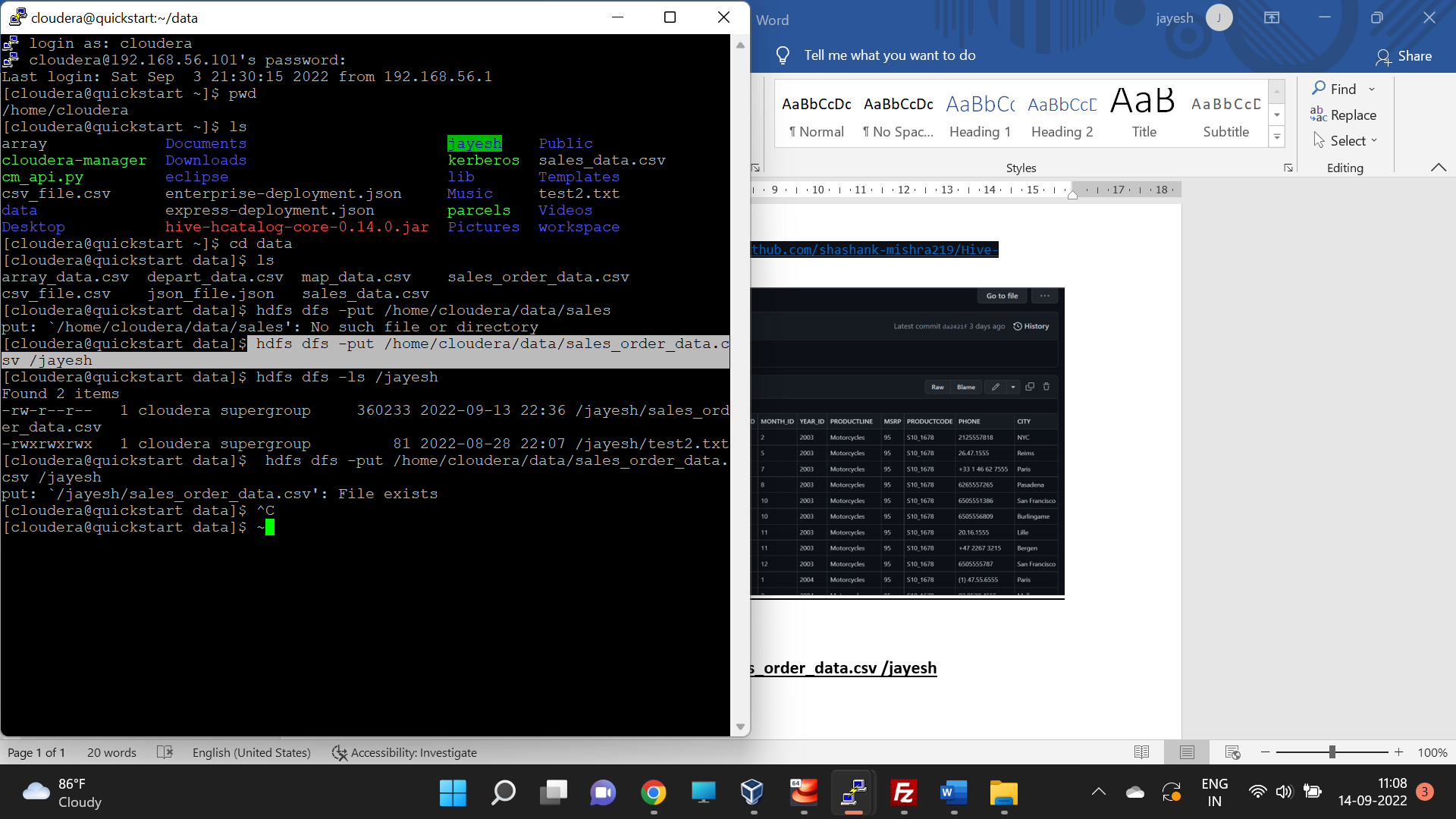
**Hive\_class\_3\_Assignment\_Solution**

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

****

1. Store raw data into hdfs location.

**hdfs dfs -put /home/cloudera/data/sales\_order\_data.csv /jayesh**

****

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.

**create table sales\_order\_data\_csv\_v1**

**(**

**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**

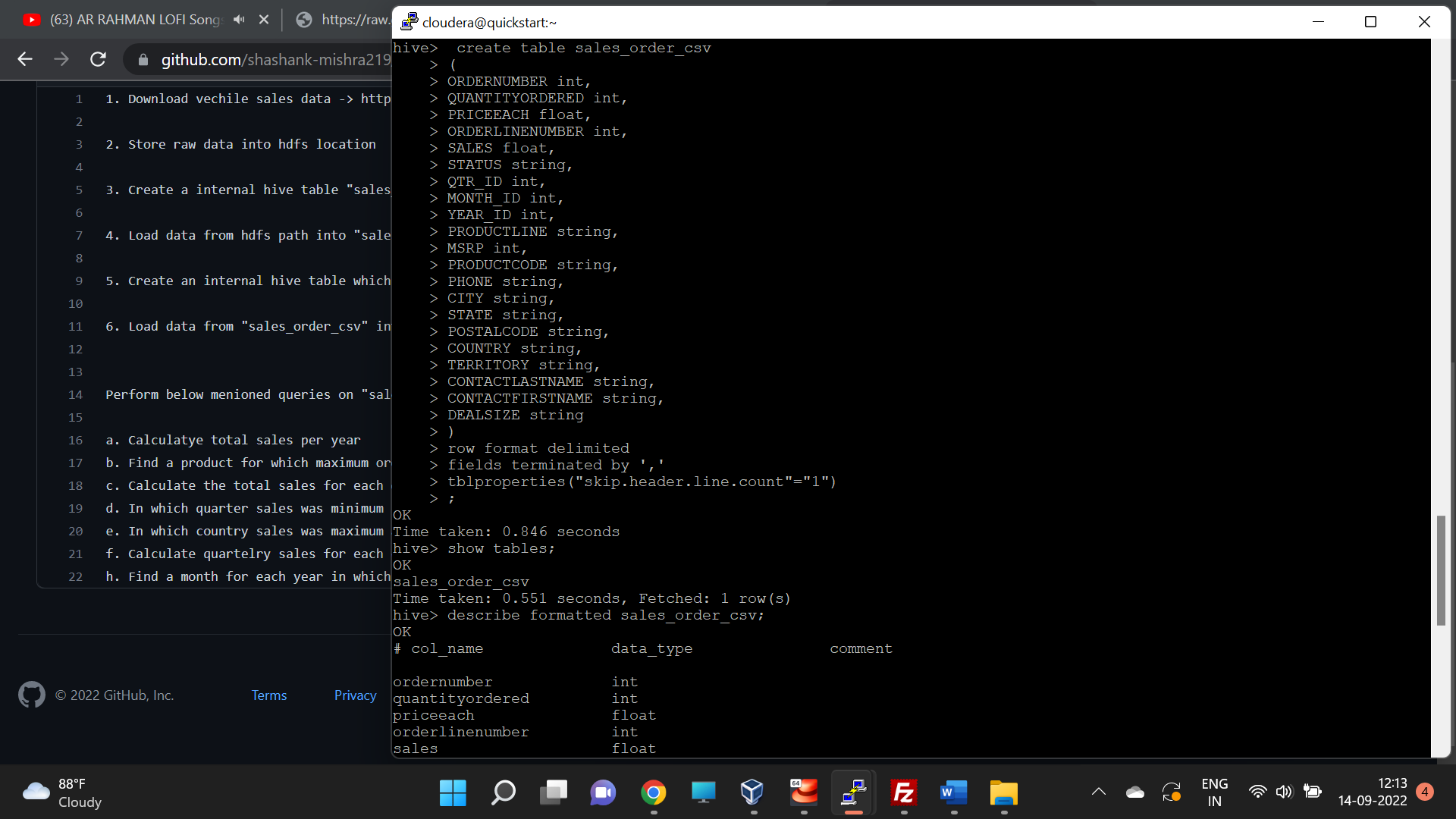
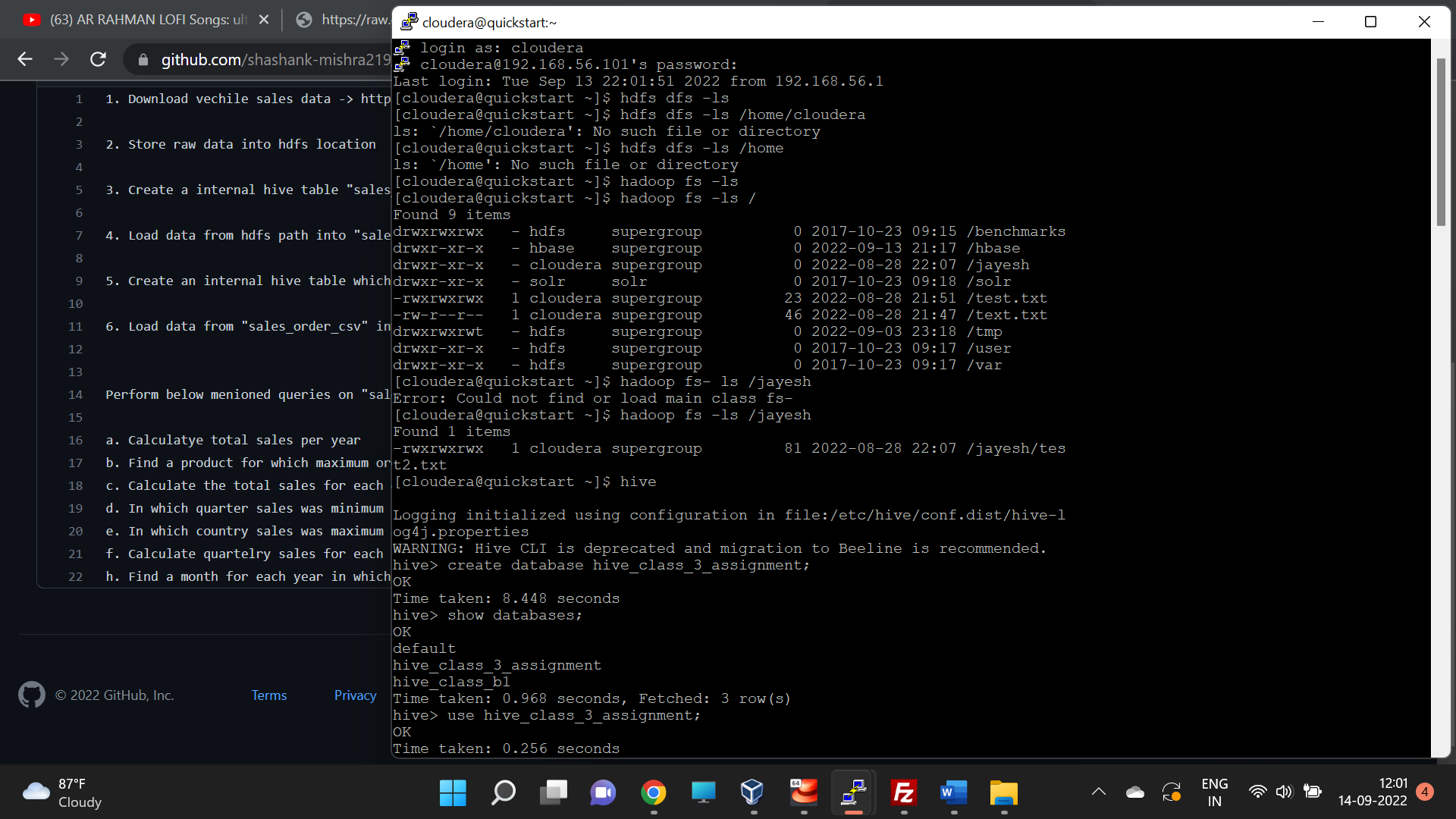
**)**

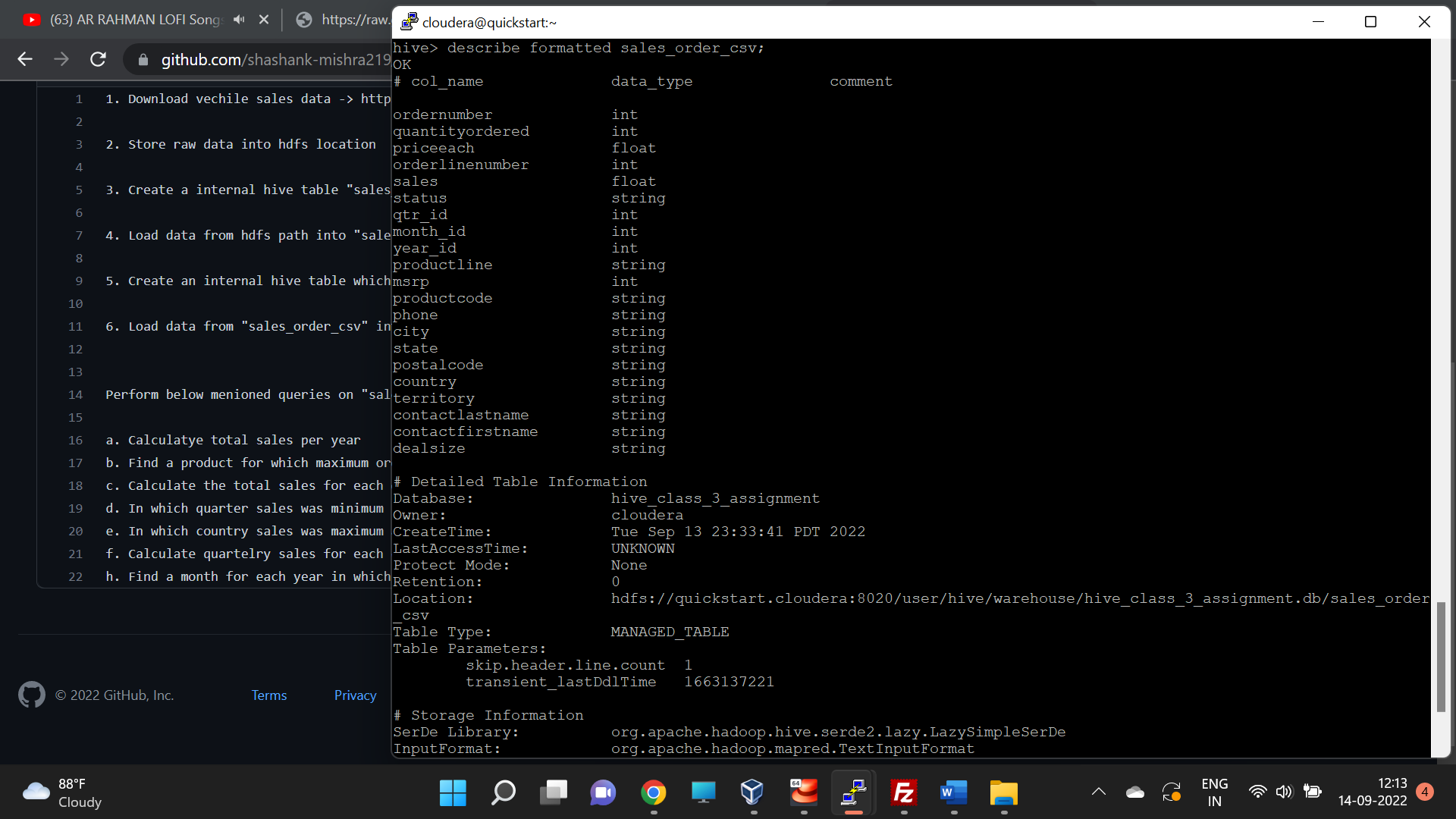
**row format delimited**

**fields terminated by ','**

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

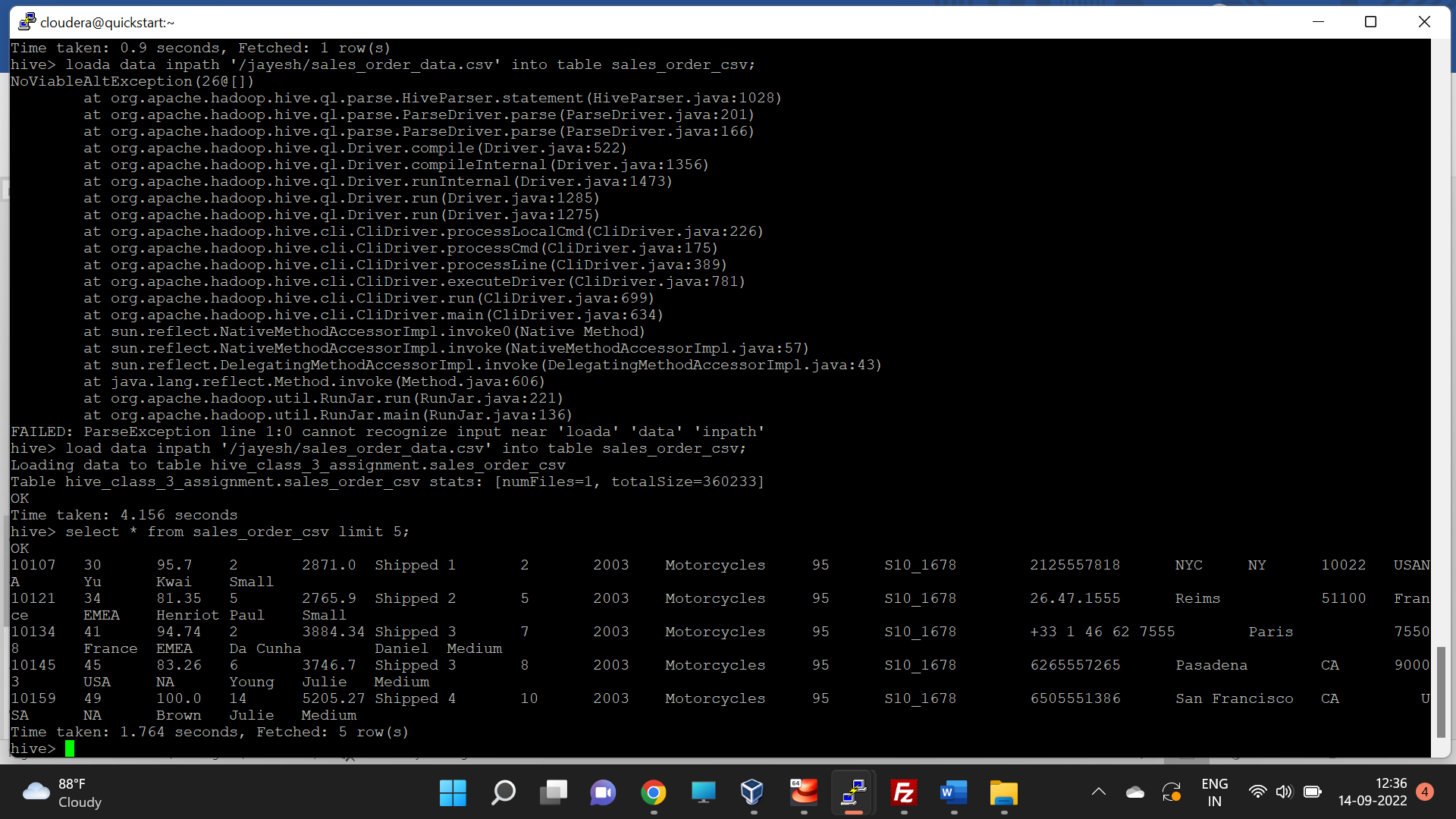
**;**

****

****

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

load data inpath '/jayesh/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\_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,**

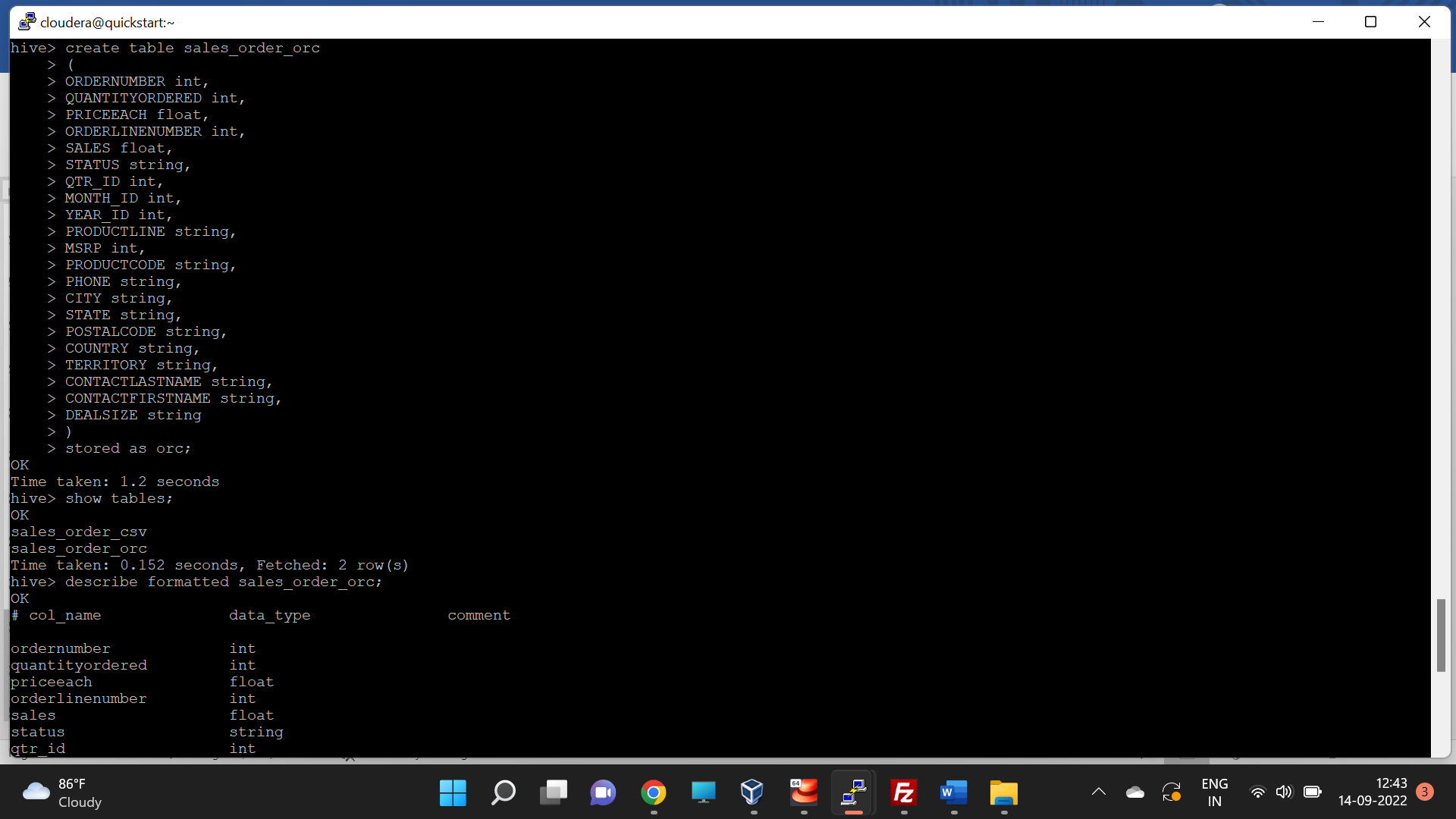
**TERRITORY string,**

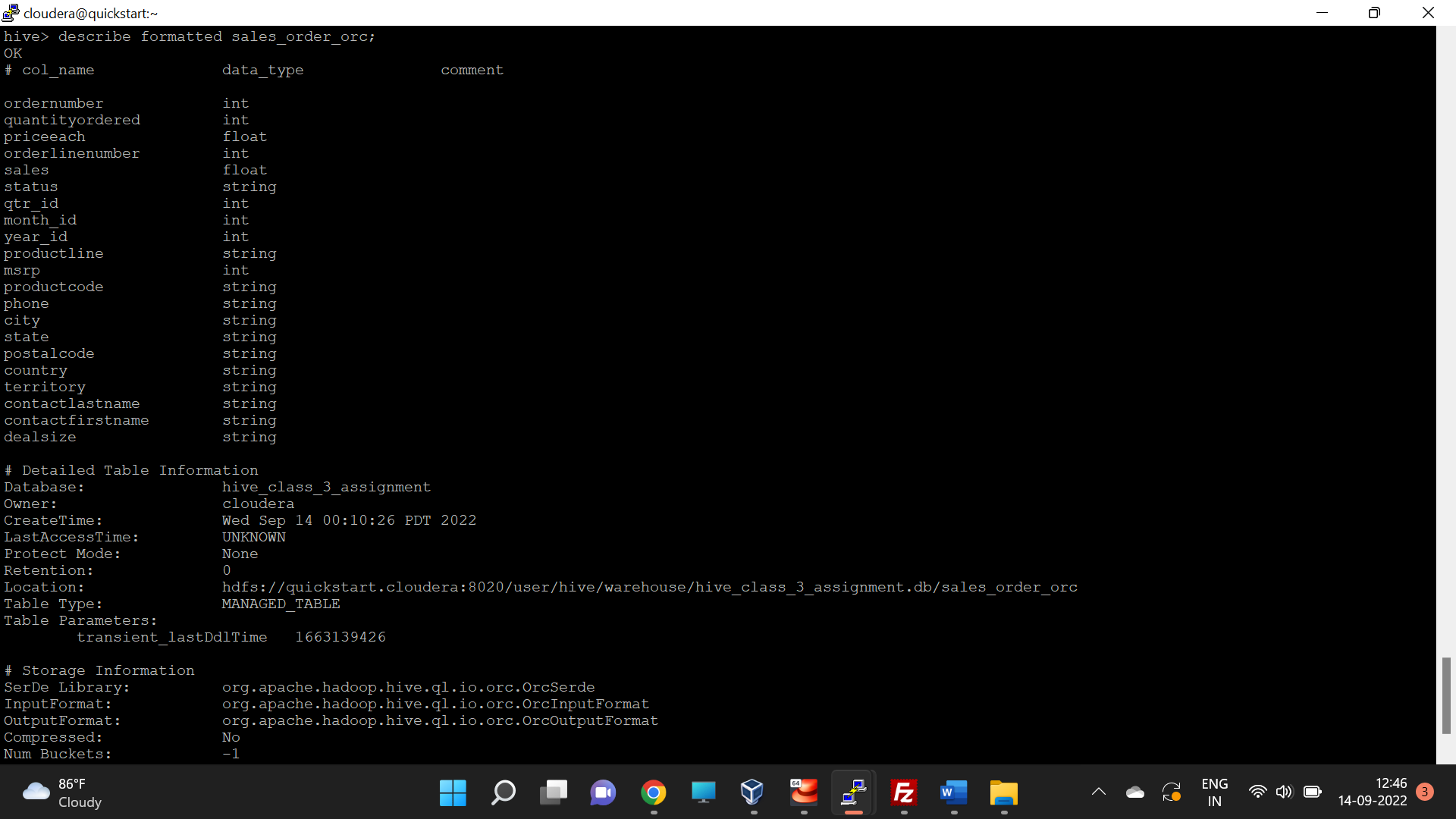
**CONTACTLASTNAME string,**

**CONTACTFIRSTNAME string,**

**DEALSIZE string**

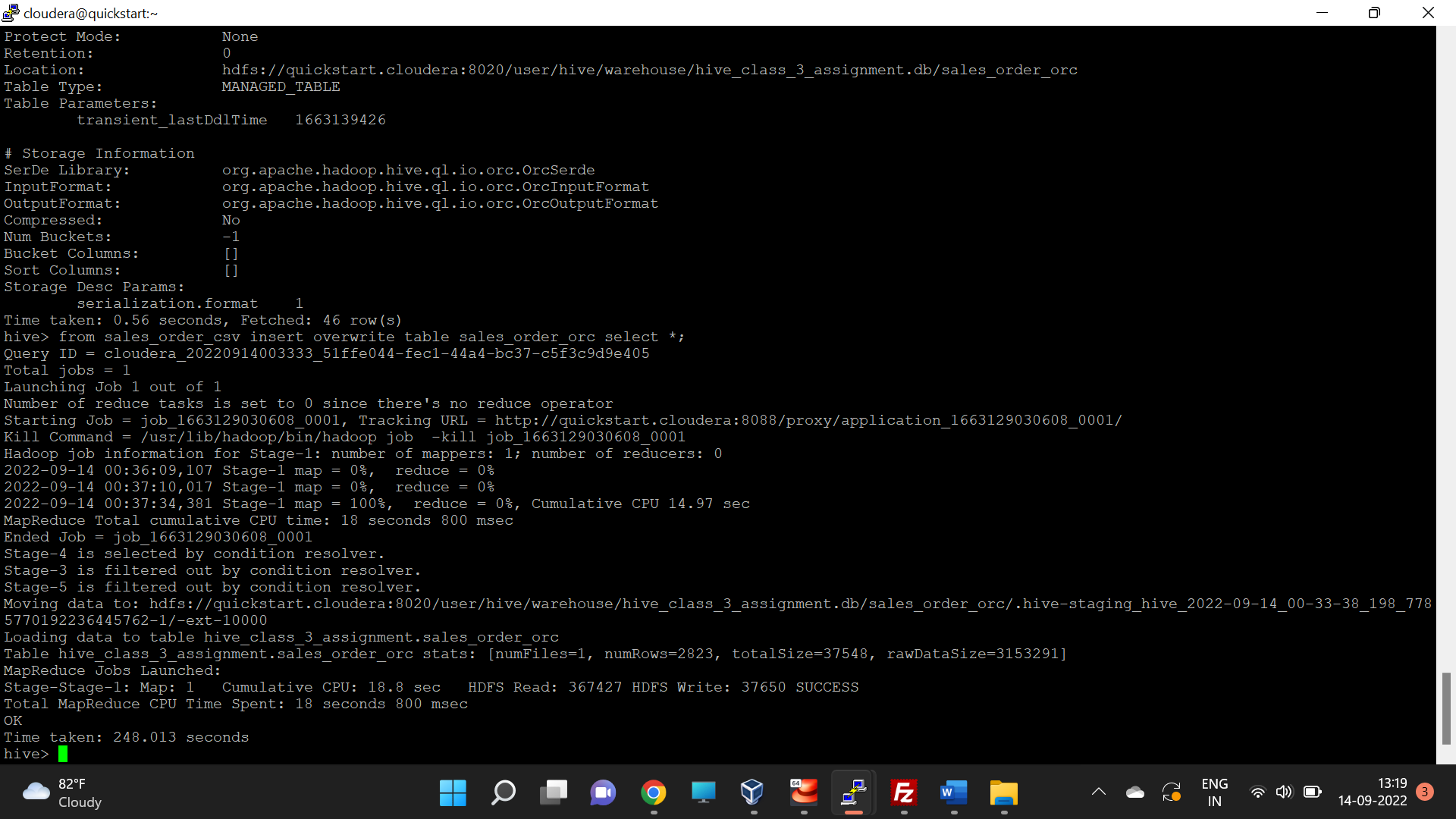
**)**

**stored as orc;** 



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

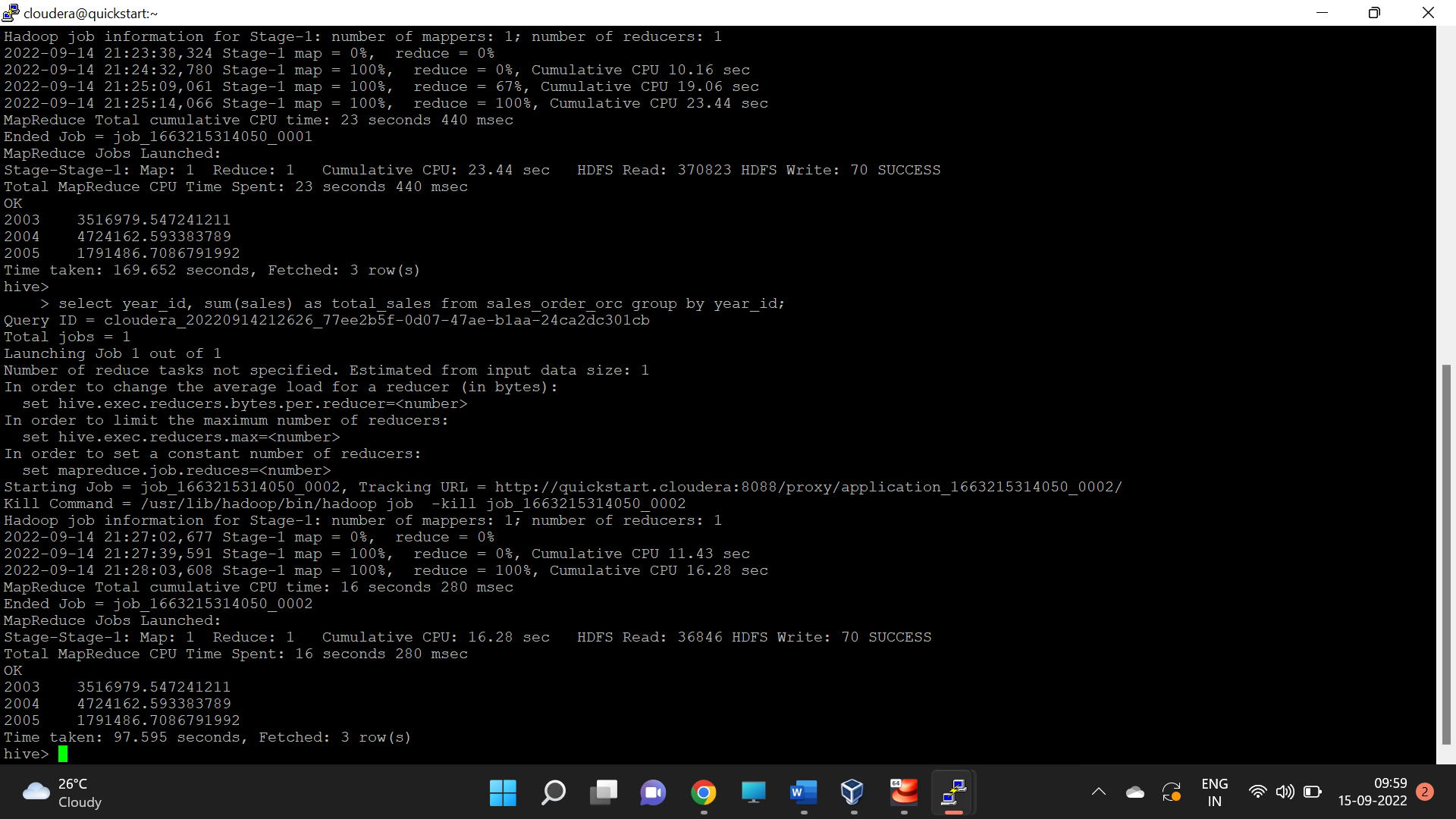
**from sales\_order\_csv insert overwrite table sales\_order\_orc select \*;**

****

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

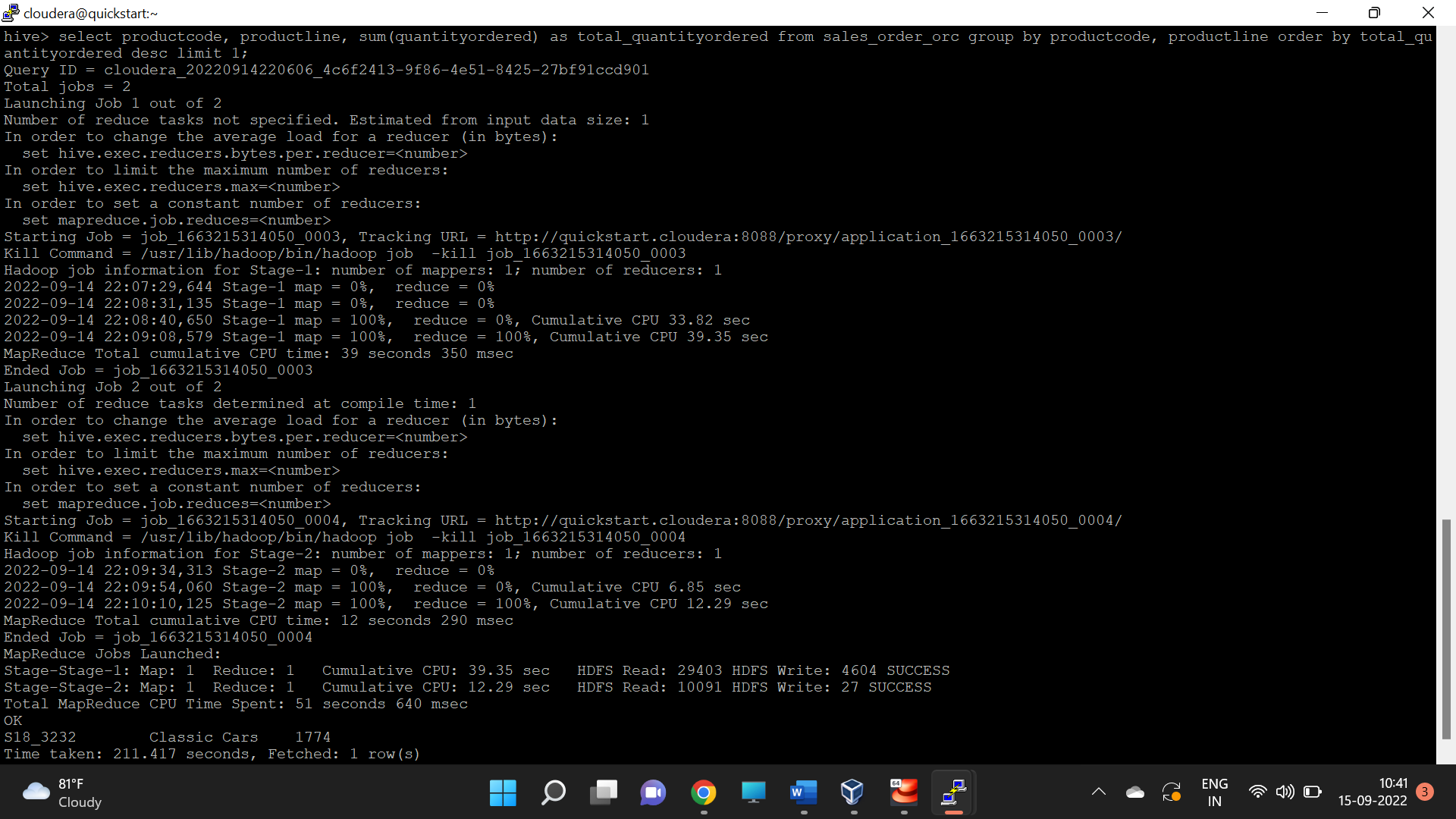
1. Calculatye total sales per year

**select year\_id, sum(sales) as total\_sales from sales\_order\_orc group by year\_id;**

****

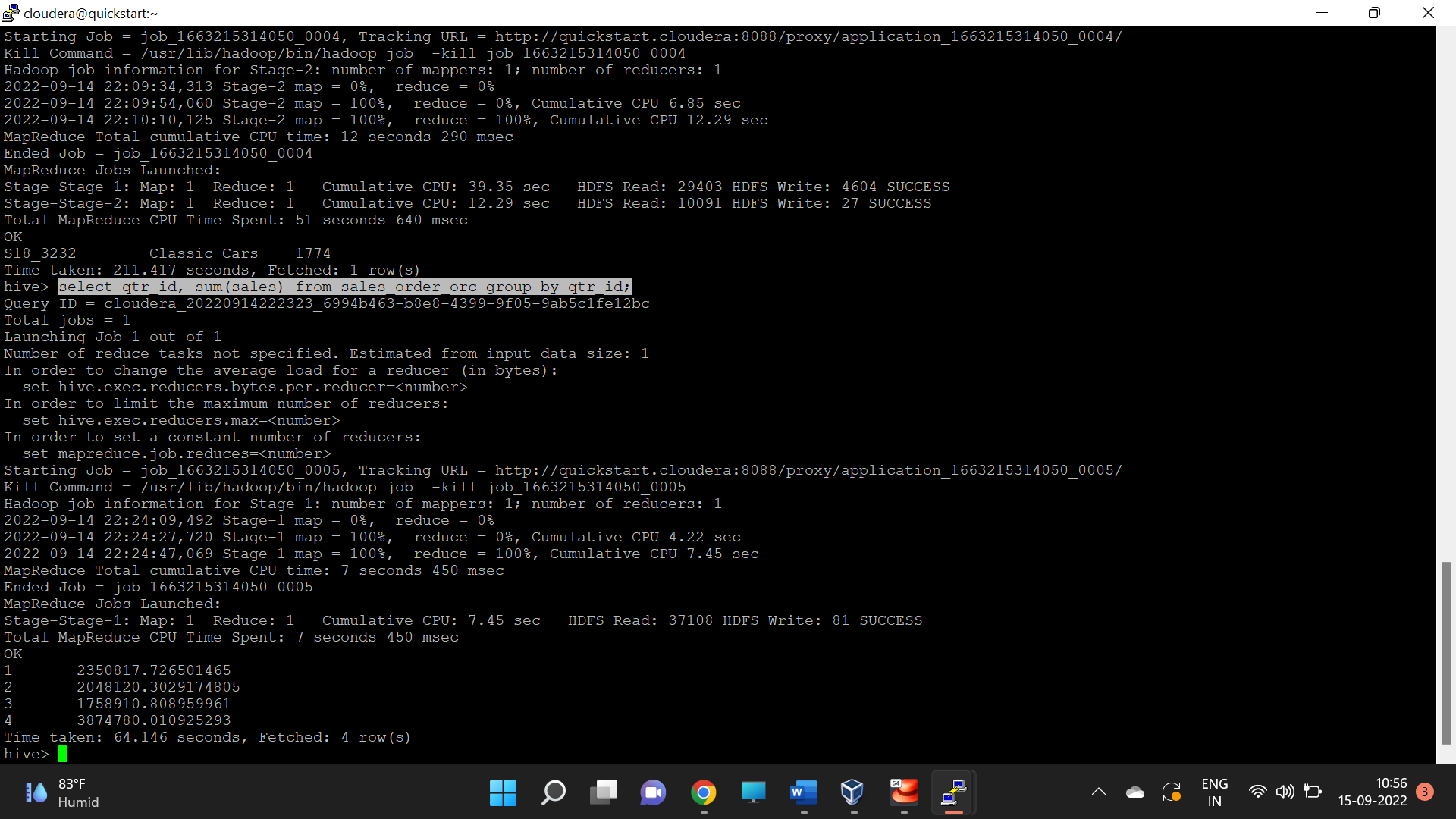
1. Find a product for which maximum orders were placed

select productcode, productline, sum(quantityordered) as total\_quantityordered from sales\_order\_orc group by productcode, productline order by total\_quantityordered desc limit 1;



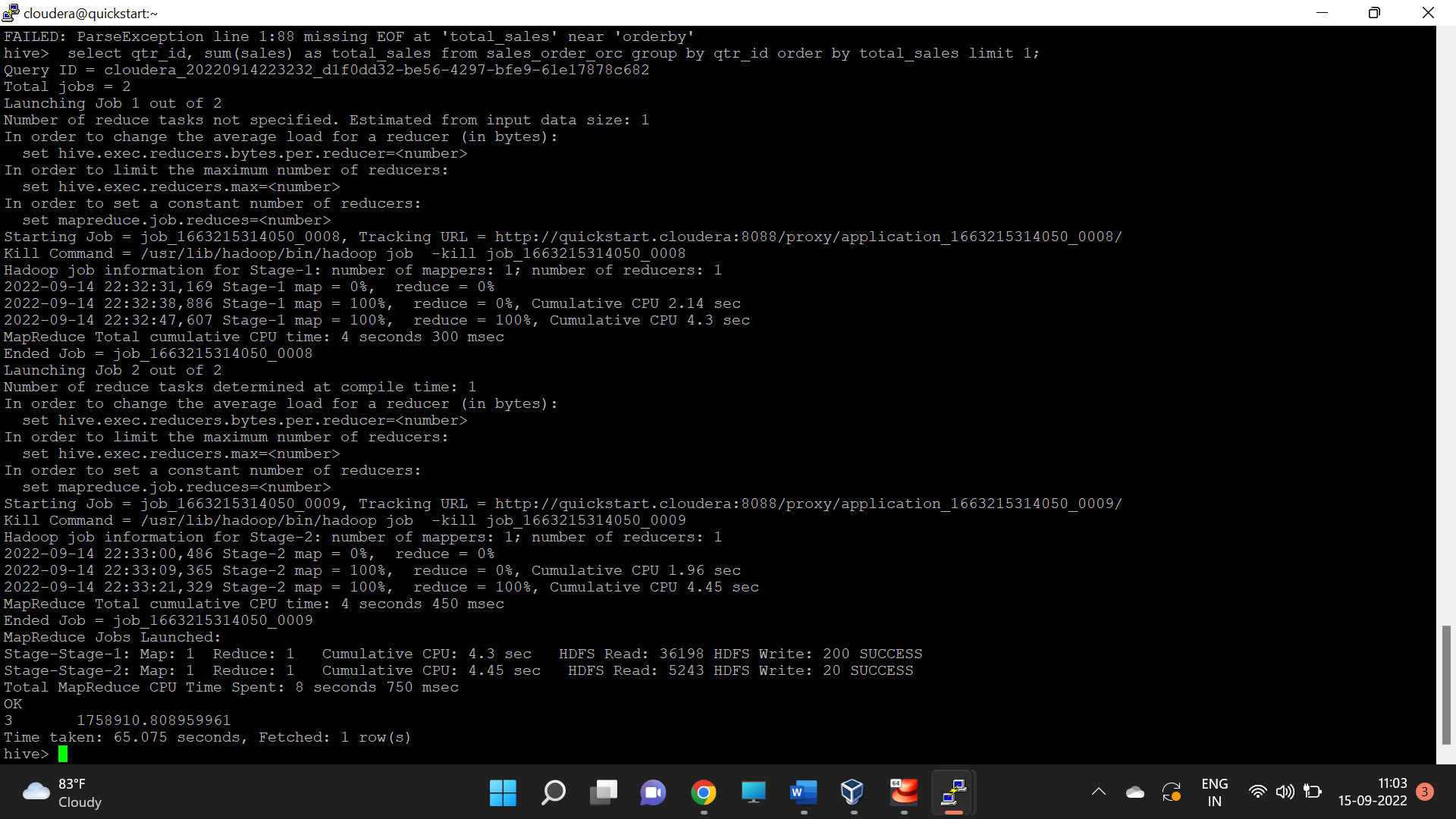
1. Calculate the total sales for each quarter

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



1. In which quarter sales was minimum

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

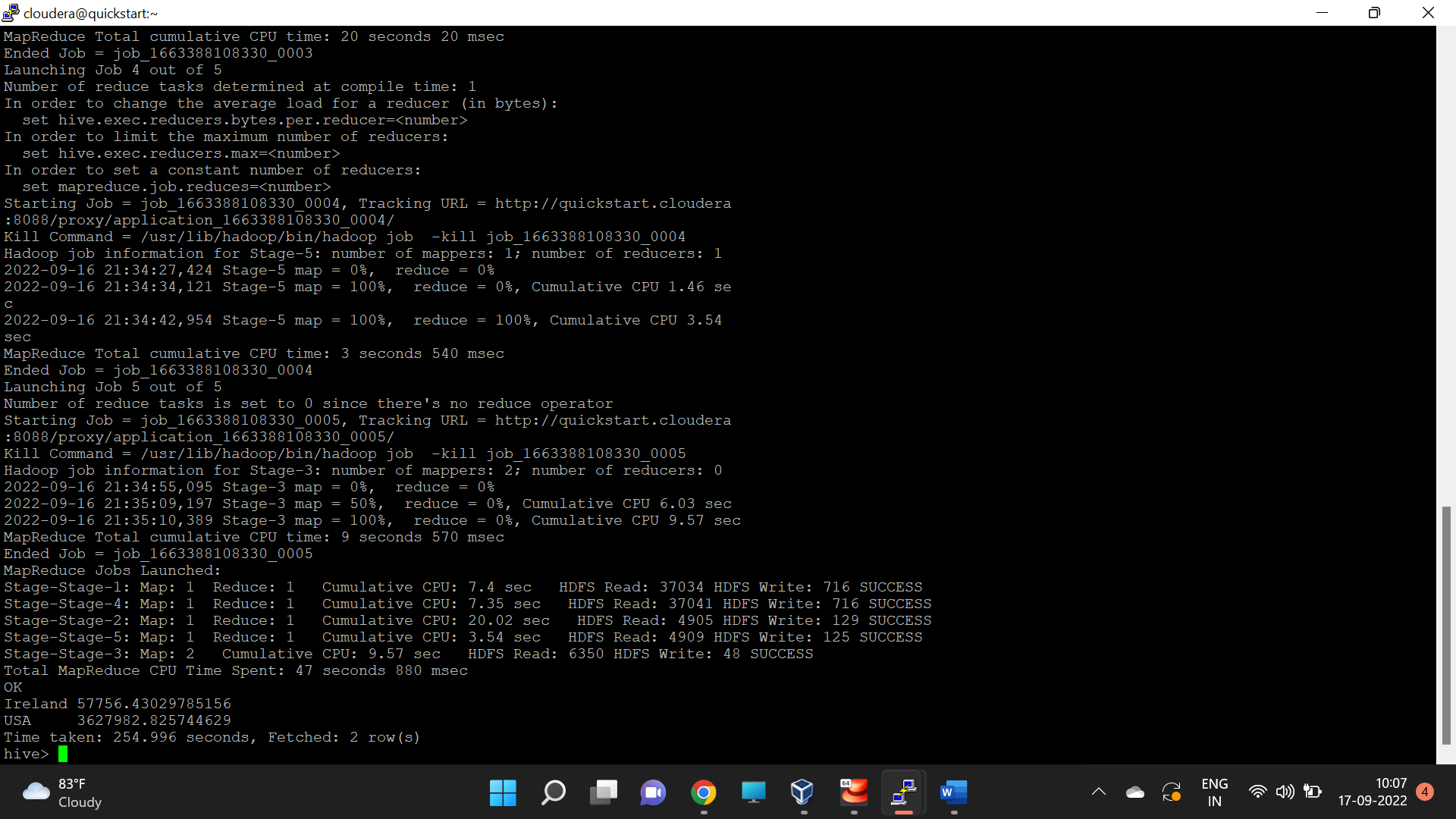


1. In which country sales was maximum and in which country sales was minimum

select country, sum(sales) as total\_sales from sales\_order\_orc group by co untry order by total\_sales limit 1

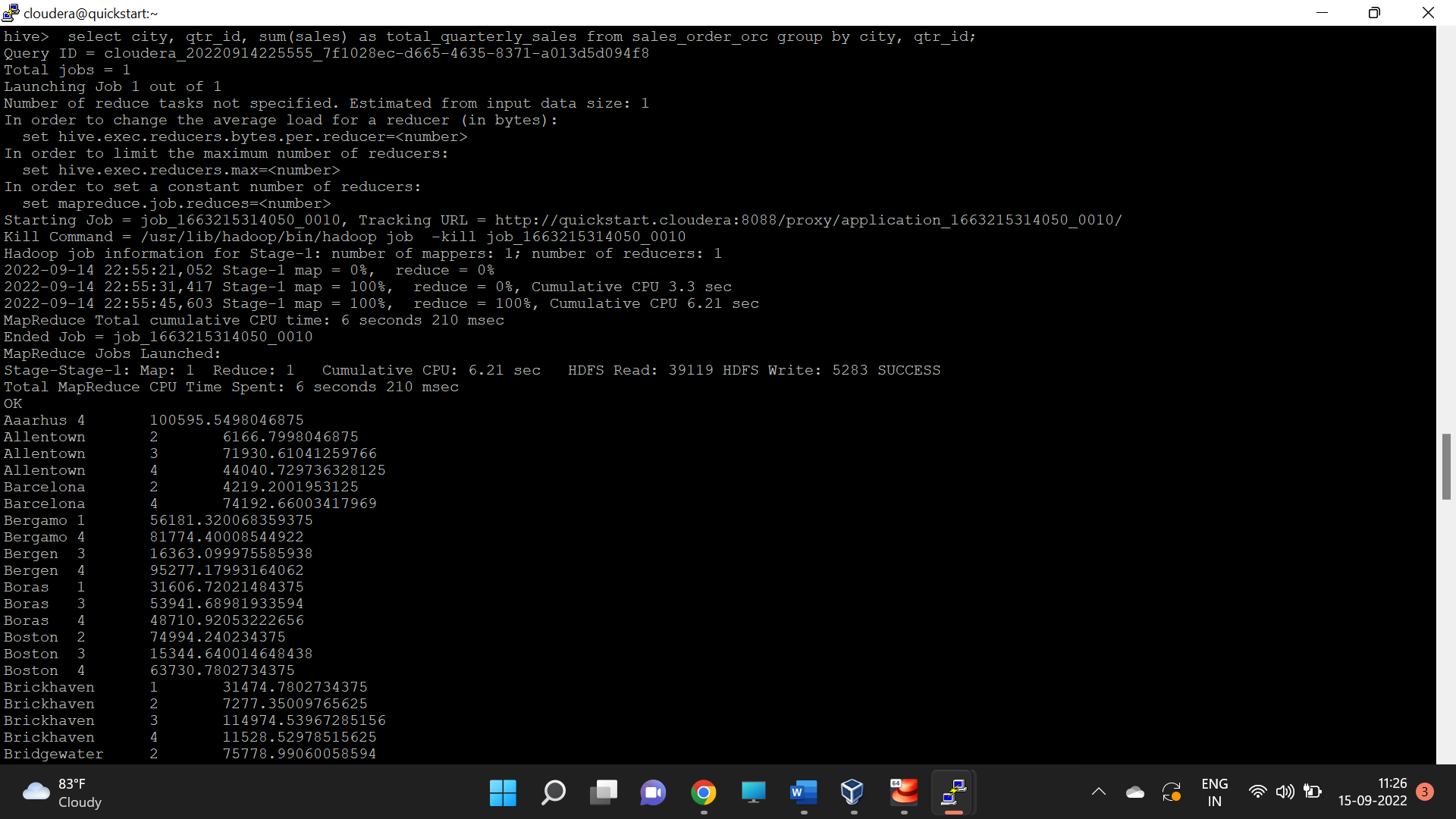
union all

select country, sum(sales) as total\_sales from sales\_order\_orc group by c ountry order by total\_sales desc limit 1;



1. Calculate quartelry sales for each city

select city, qtr\_id, sum(sales) as total\_quarterly\_sales from sales\_order\_orc group by city, qtr\_id;



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

select year\_id,month\_id,total\_quantityordered from (select year\_id, month\_id, total\_quantityordered, dense\_rank() over (partition by year\_id order by total\_quantityordered desc) as dns from (select year\_id, month\_id, sum(quantityordered) as total\_quantityordered from sales\_order\_csv group by year\_id, month\_id)t1)t2 where dns = 1;

