**hive\_class\_3\_assignment**

**Name – Ayushi Jain**

**Batch – 1st Big data bootcamp**

**Mail Id –** [**ayushijain2277@gmail.com**](mailto:ayushijain2277@gmail.com)

1. -- **Downloaded file and moved to /tmp/Hive\_class/ folder in local.**

2. --**Storing raw data into hdfs location**

hadoop fs -put sales\_order\_data.csv /tmp/hive\_data\_class2

Hadoop fs -ls /tmp/hive\_data\_class2

1. --**Creating a internal hive table "sales\_order\_csv" which will store csv data sales\_order\_csv and making sure to skip header row while creating table**

use hive\_class\_b1;

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

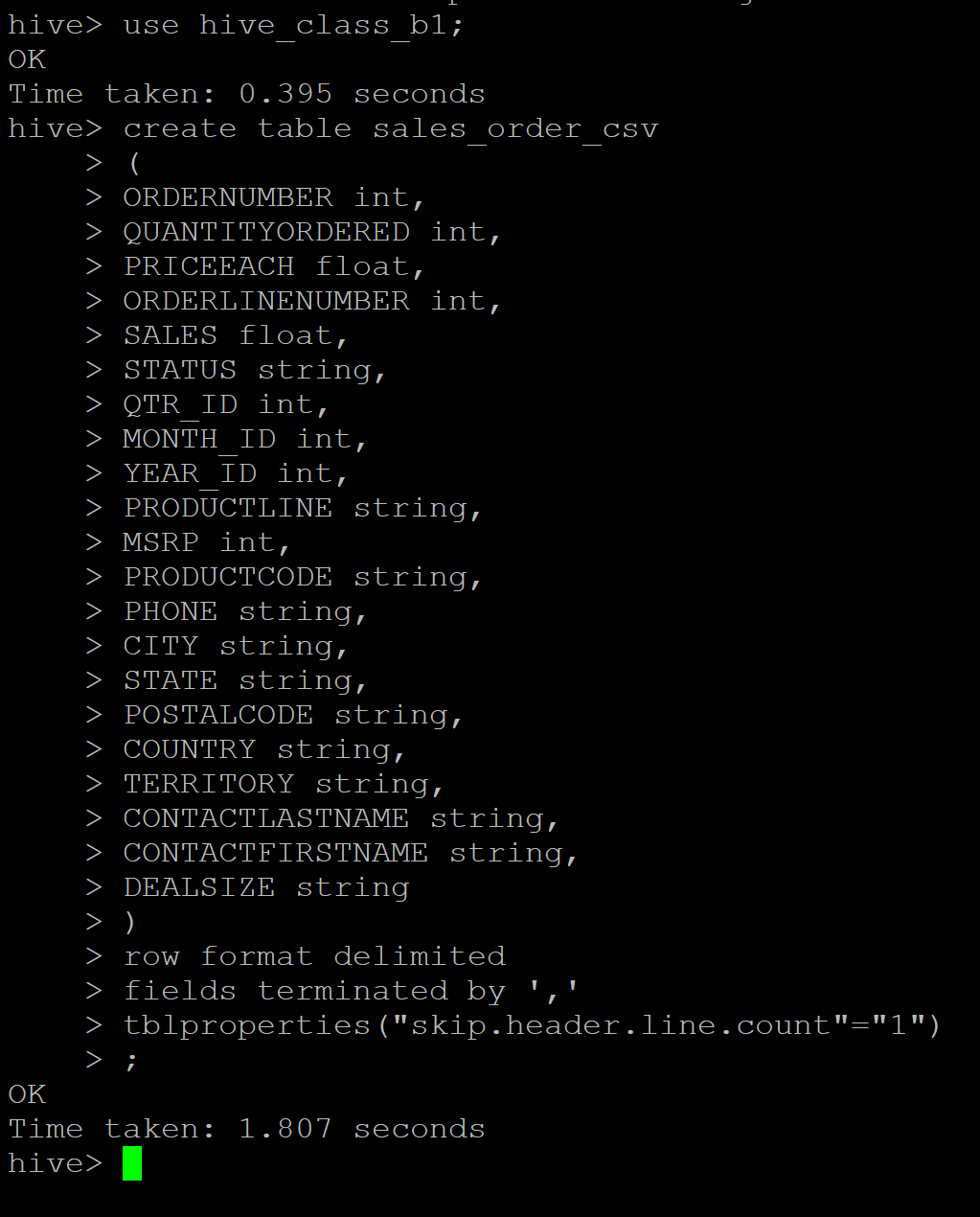
)

row format delimited

fields terminated by ','

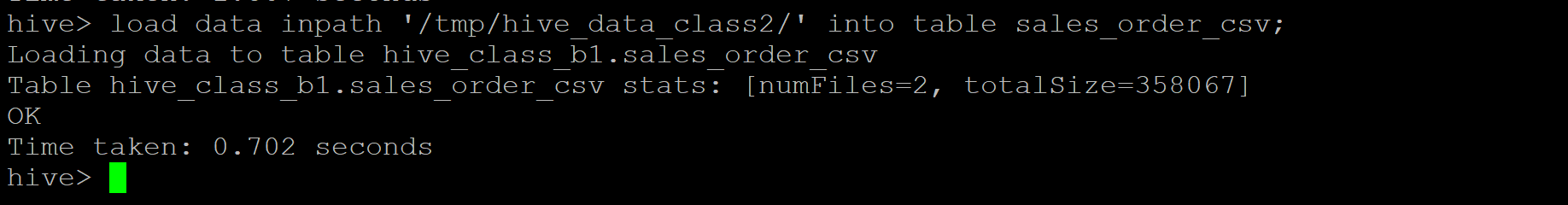
tblproperties("skip.header.line.count"="1")

;



4.--**Loading data from hdfs path into "sales\_order\_csv" internal table**

Load data inpath ‘/tmp/hive\_data\_class2/’ into table sales\_order\_csv;



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

create table sales\_order\_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

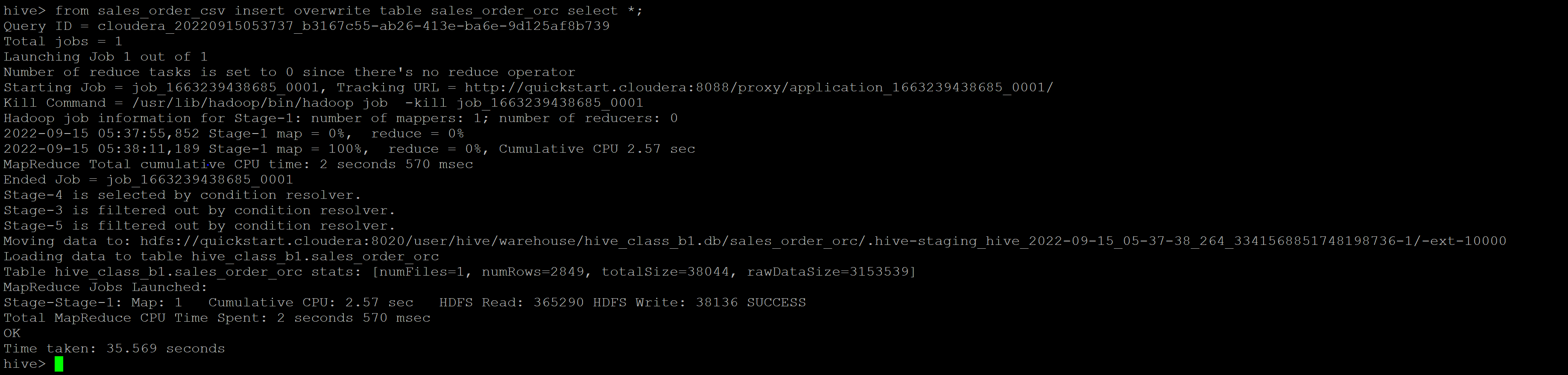
)

stored as orc;



1. --**Loading data from "sales\_order\_csv" into "sales\_order\_orc"**

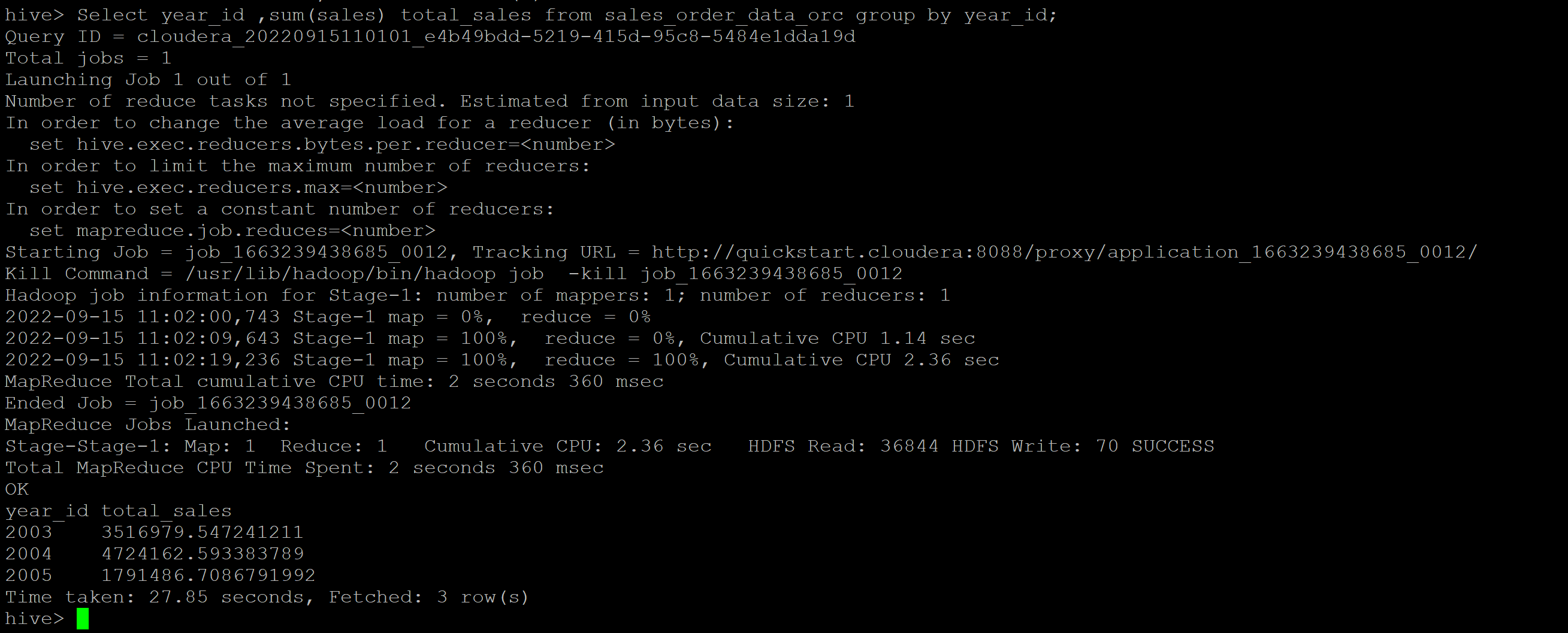
from sales\_order\_csv insert overwrite table sales\_order\_orc select \*;



**QUERIES :**

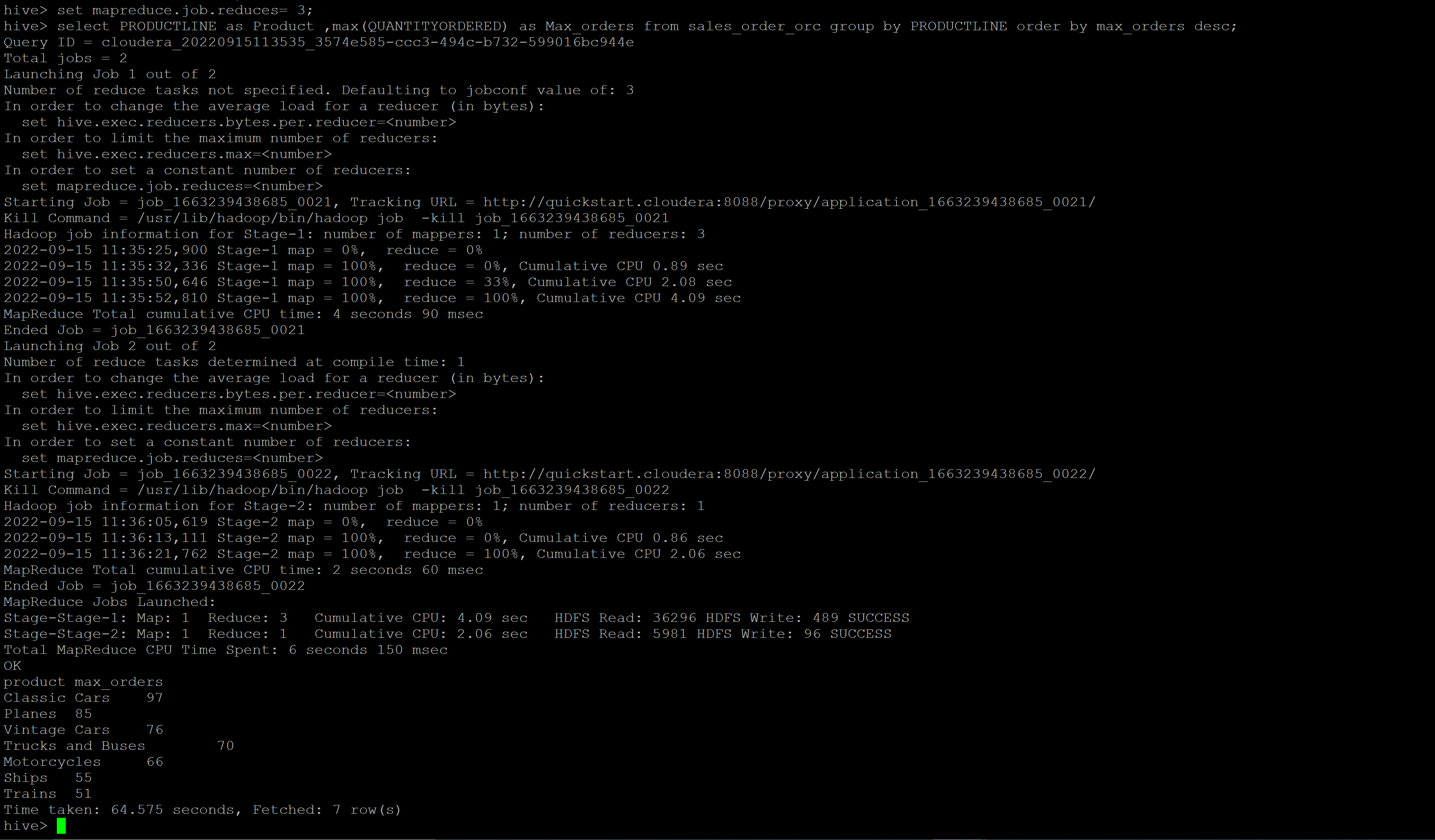
1. **Calculatye total sales per year**

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



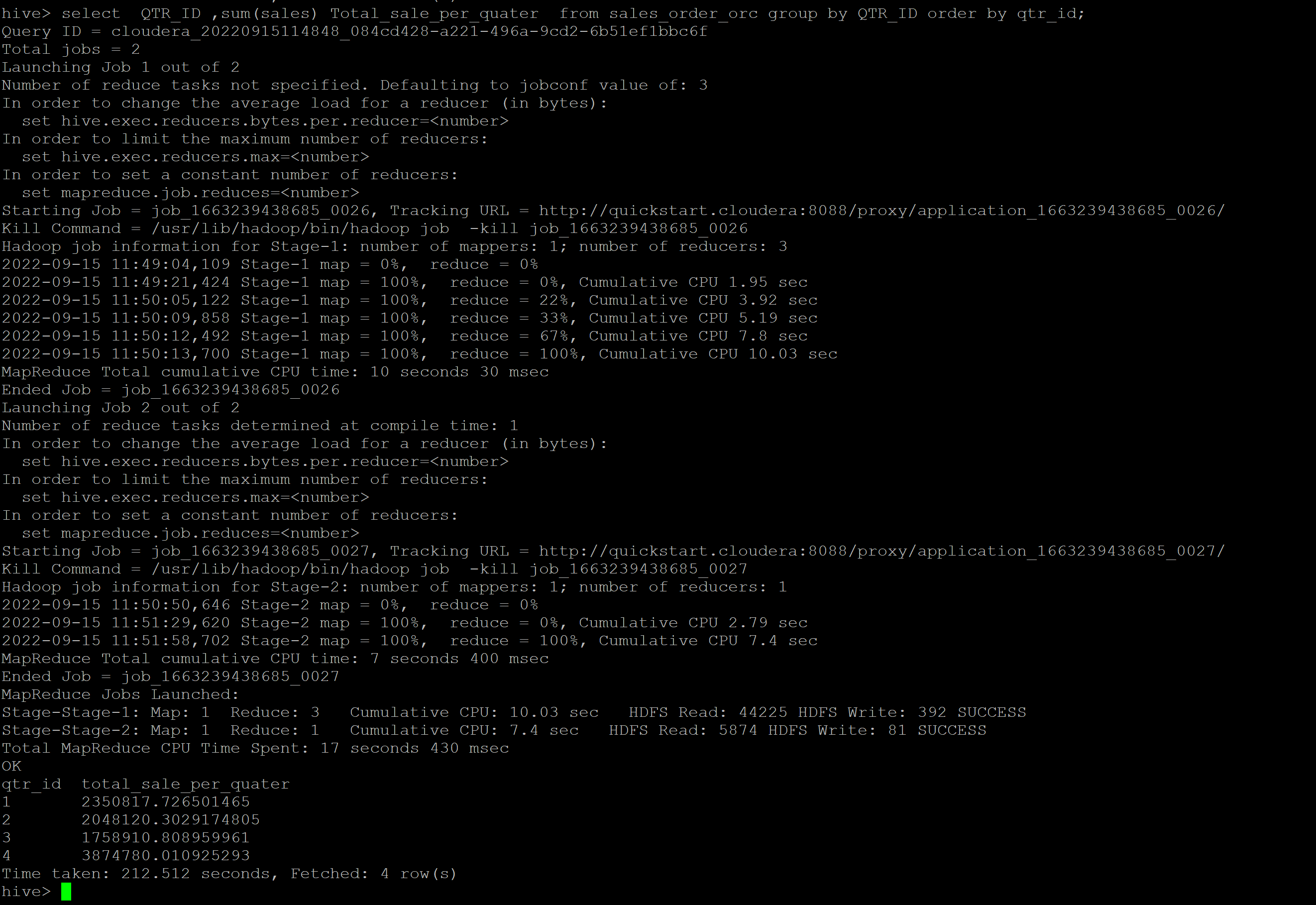
1. **Find a product for which maximum orders were placed**

select PRODUCTLINE as Product ,max(QUANTITYORDERED) as Max\_orders from sales\_order\_orc group by PRODUCTLINE order by max\_orders desc;



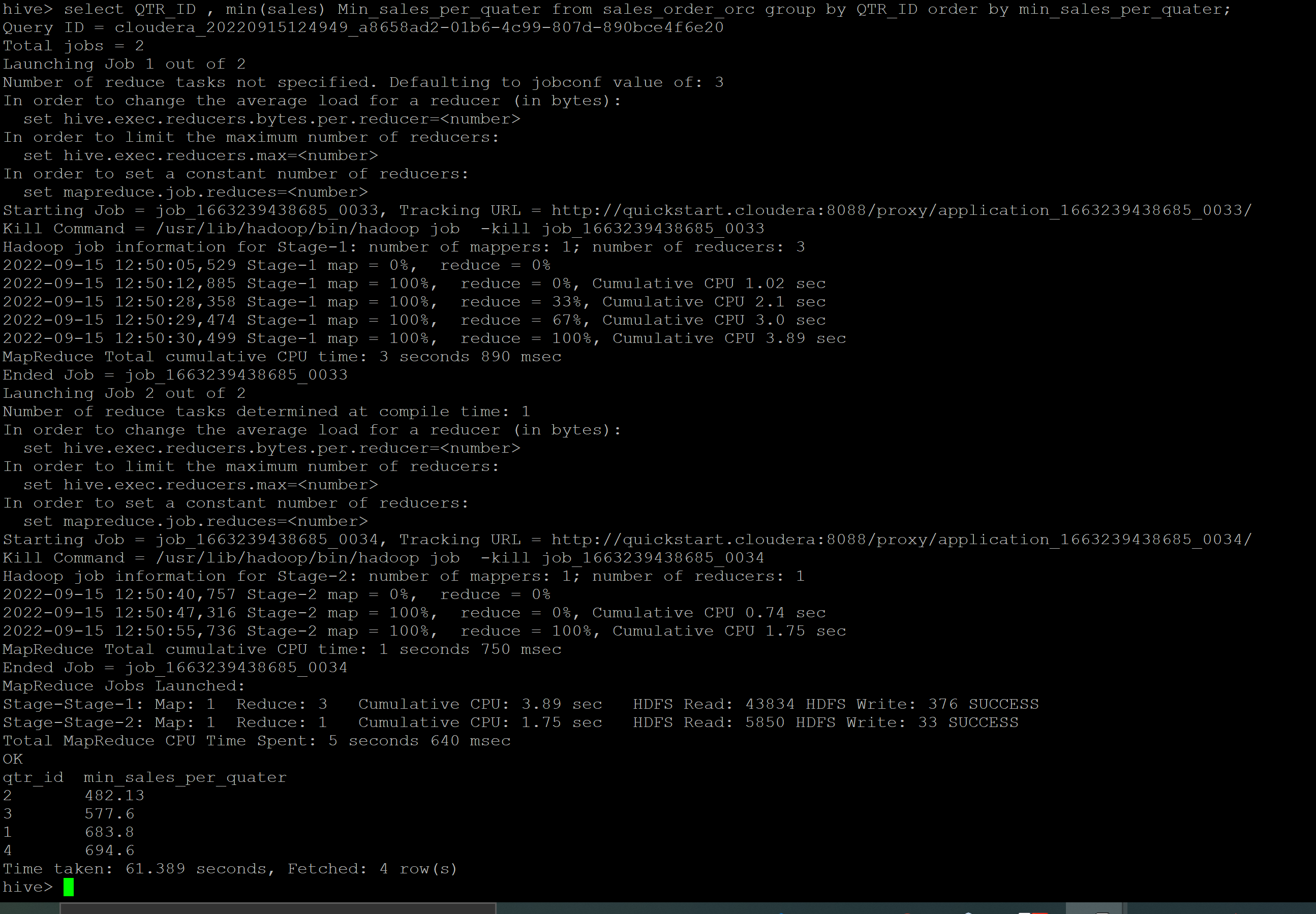
1. **Calculate the total sales for each quarter**

select QTR\_ID ,sum(sales) Total\_sale\_per\_quater from sales\_order\_orc group by QTR\_ID order by qtr\_id;



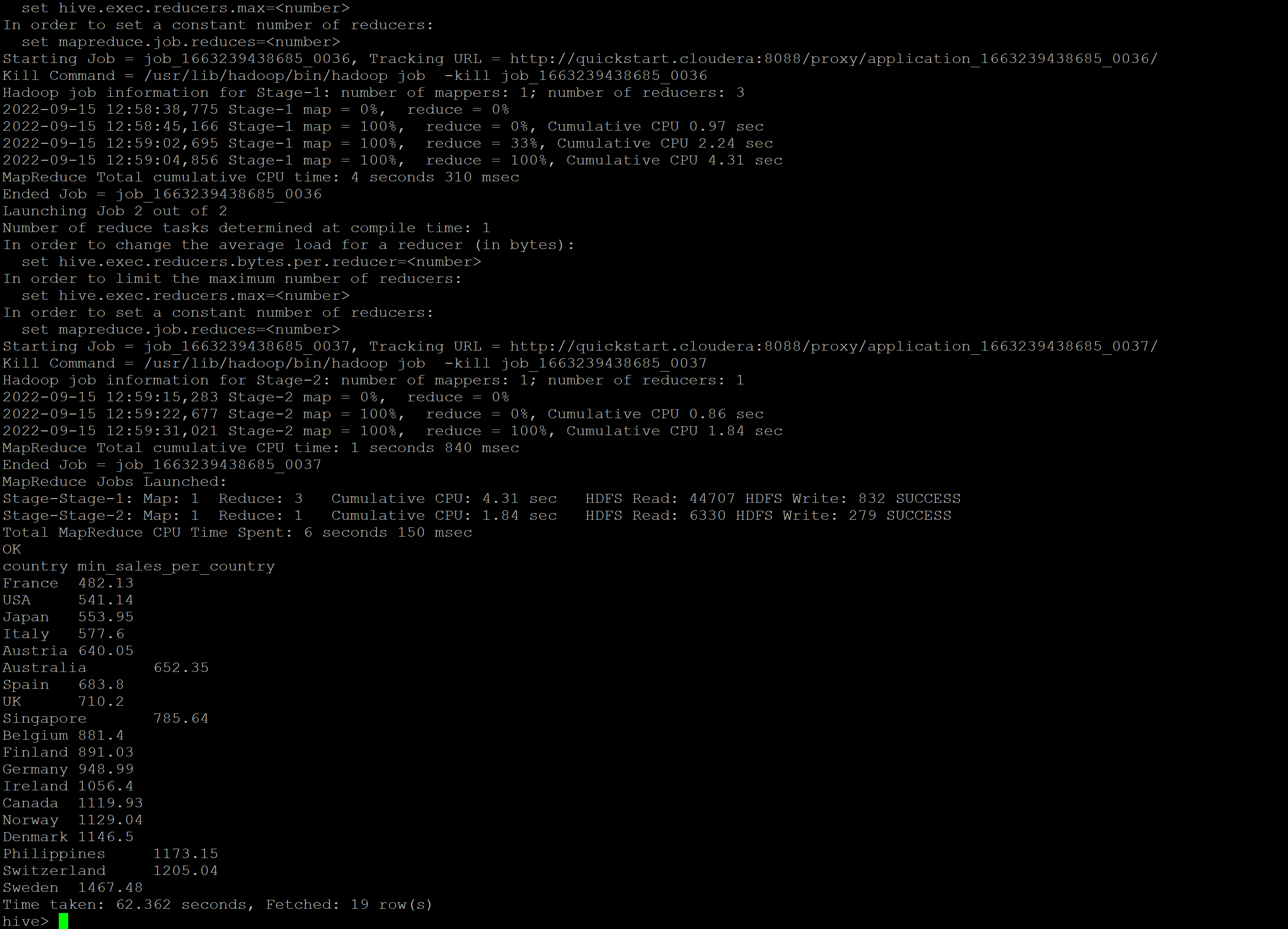
1. **In which quarter sales was minimum**

select QTR\_ID , min(sales) Min\_sales\_per\_quater from sales\_order\_orc group by QTR\_ID order by min\_sales\_per\_quater;

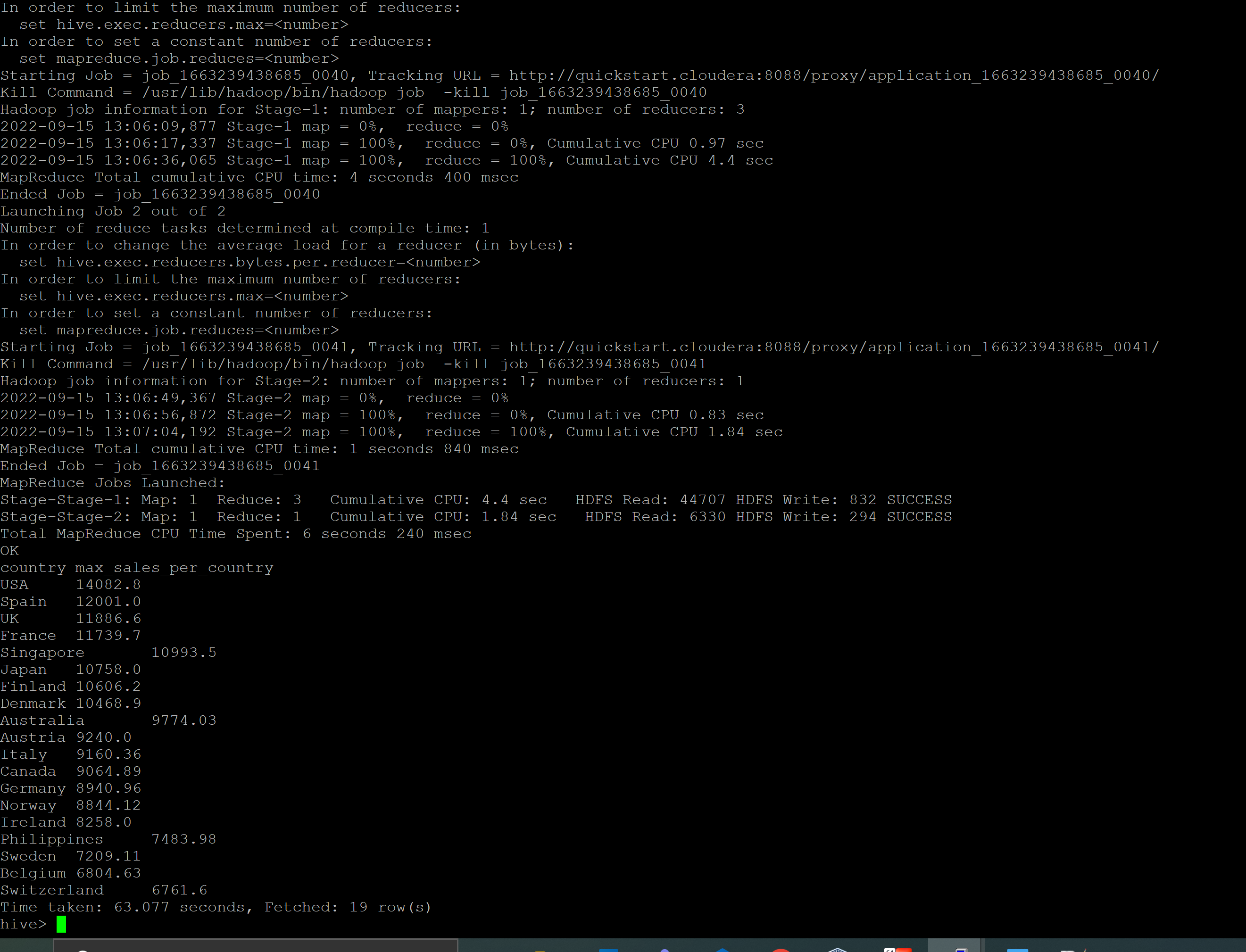


1. **In which country sales was maximum and in which country sales was minimum**

select COUNTRY , min(sales) Min\_sales\_per\_country from sales\_order\_orc group by COUNTRY order by Min\_sales\_per\_country;



select COUNTRY , max(sales) Max\_sales\_per\_country from sales\_order\_orc group by COUNTRY order by Max\_sales\_per\_country desc;



1. **Calculate quartelry sales for each city**

select CITY ,QTR\_ID , sum(sales) Quaterly\_sales\_per\_city from sales\_order\_orc group by QTR\_ID ,CITY ORDER BY QTR\_ID ,CITY;

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

SELECT YEAR\_ID , MONTH\_ID , MAX(QUANTITYORDERED) FROM sales\_order\_orc group by MONTH\_ID , YEAR\_ID order by YEAR\_ID , MONTH\_ID;

