**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"**

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

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

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

**a. Calculate total sales per year**

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

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

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

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

**f. Calculate quarterly sales for each city**

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

# Solutions:-

# create csv table for sales data

**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 varchar(50),**

**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")** **;**

**load data local inpath 'file:///config/workspace/sales\_order\_data.csv' into table sales\_order\_csv;**

# load sales\_order\_data.csv data into above mentioned tables

# 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 varchar(50),

# CITY string,

# STATE string,

# POSTALCODE int,

# COUNTRY string,

# TERRITORY string,

# CONTACTLASTNAME string,

# CONTACTFIRSTNAME string,

# DEALSIZE string

# )

# stored as orc;

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

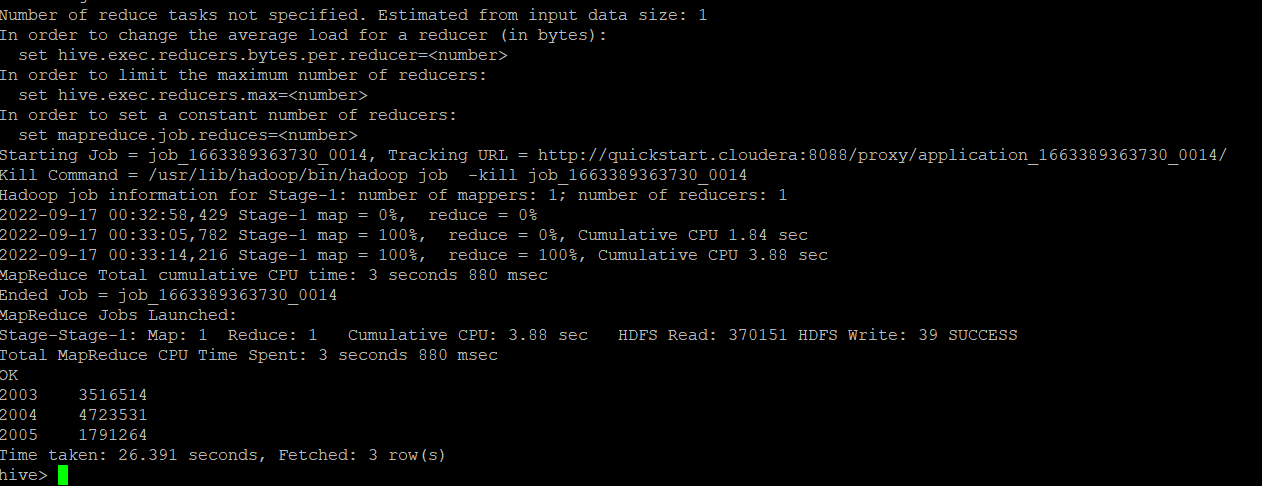
**Set column header**

set hive.cli.print.header = true;

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

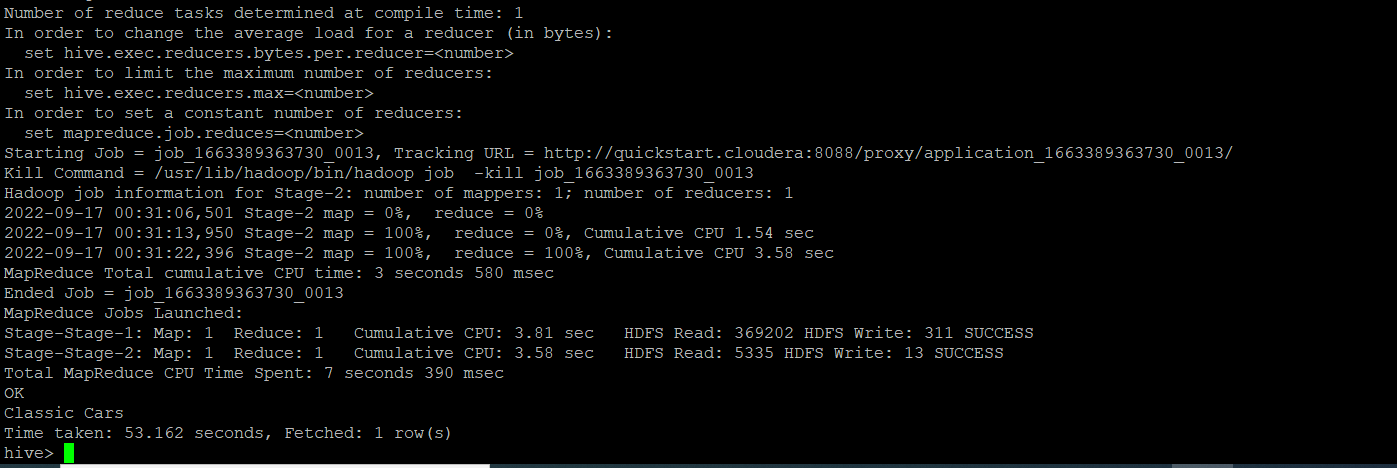
**a. Calculate total sales per year**

Ans:- **hive> select year\_ID as year,sum(sales) as Total\_Sales from sales\_order\_csv group by year\_ID;**



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

Ans:- **hive> select PRODUCTLINE from (select PRODUCTLINE ,sum(QUANTITYORDERED) max from sales\_order\_csv group by PRODUCTLINE order by max desc limit 1) a ;**



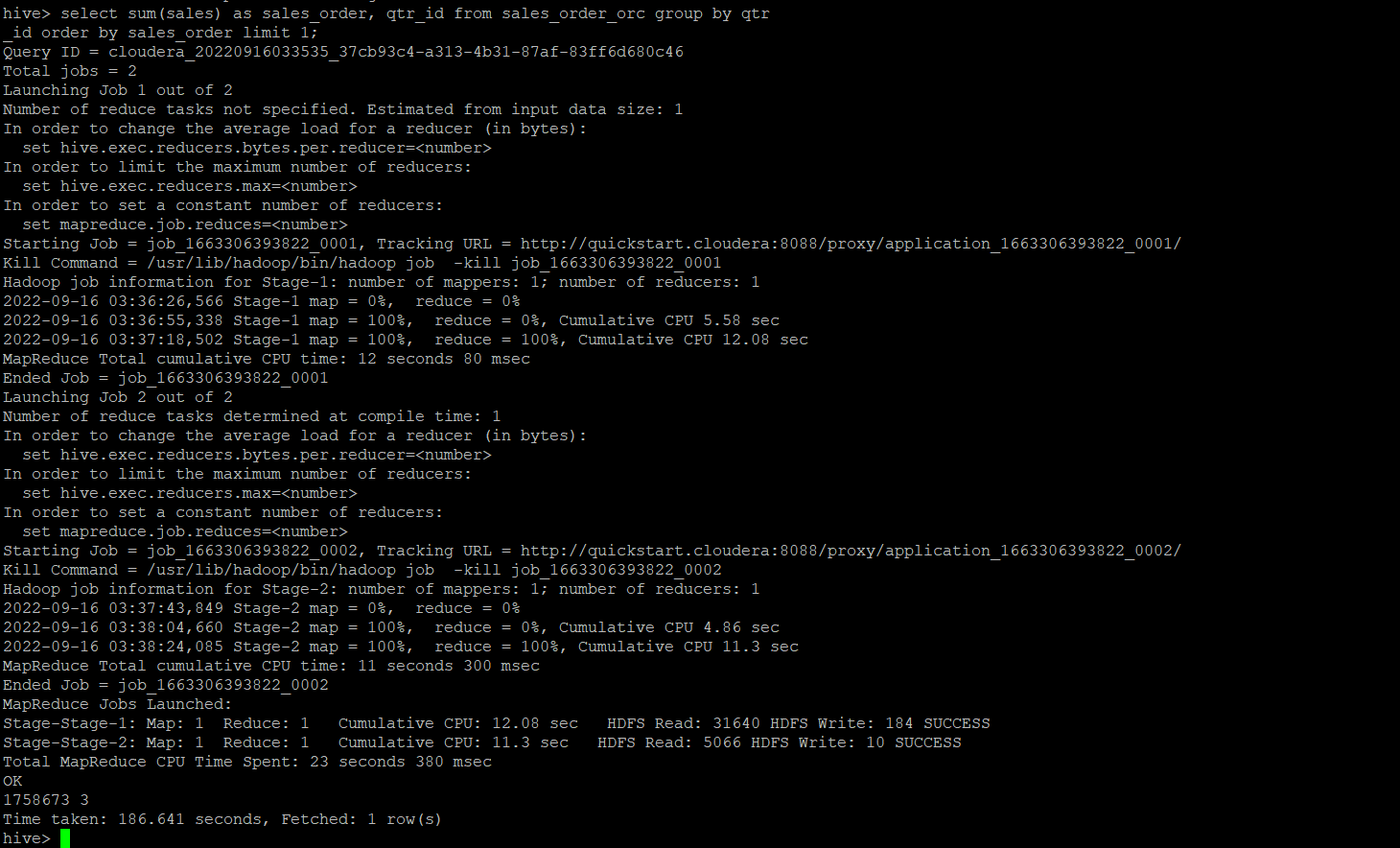
**c. Calculate the total sales for each quarter**

Ans:- **select sum(sales) as Total\_sales,QTR\_ID from sales\_order\_csv group by QTR\_ID order by QTR\_ID;**

# 

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

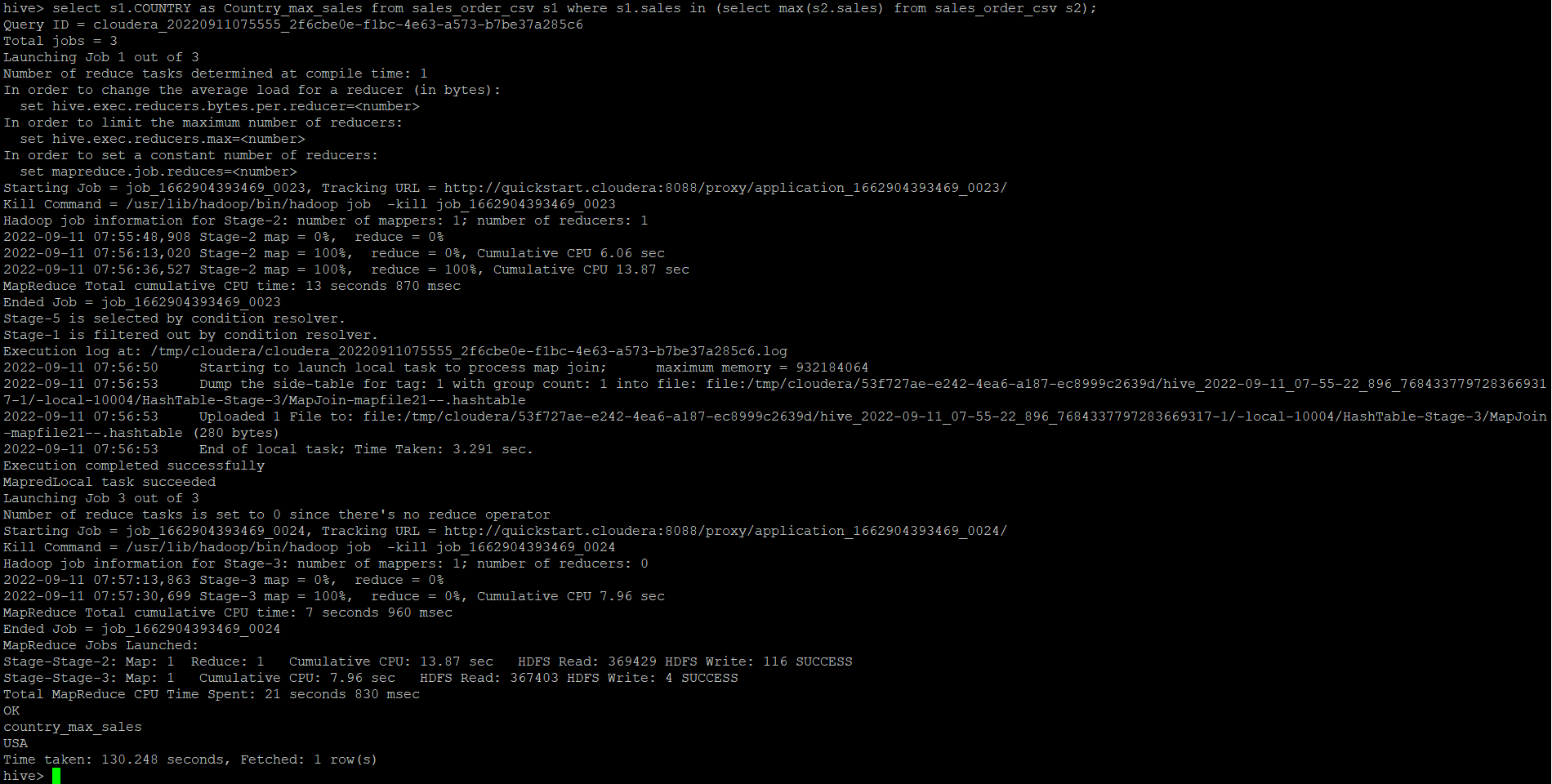
Ans:- **hive>select sum(sales) as sales\_order, qtr\_id from sales\_order\_orc group by qtr\_id order by sales\_order limit 1;**

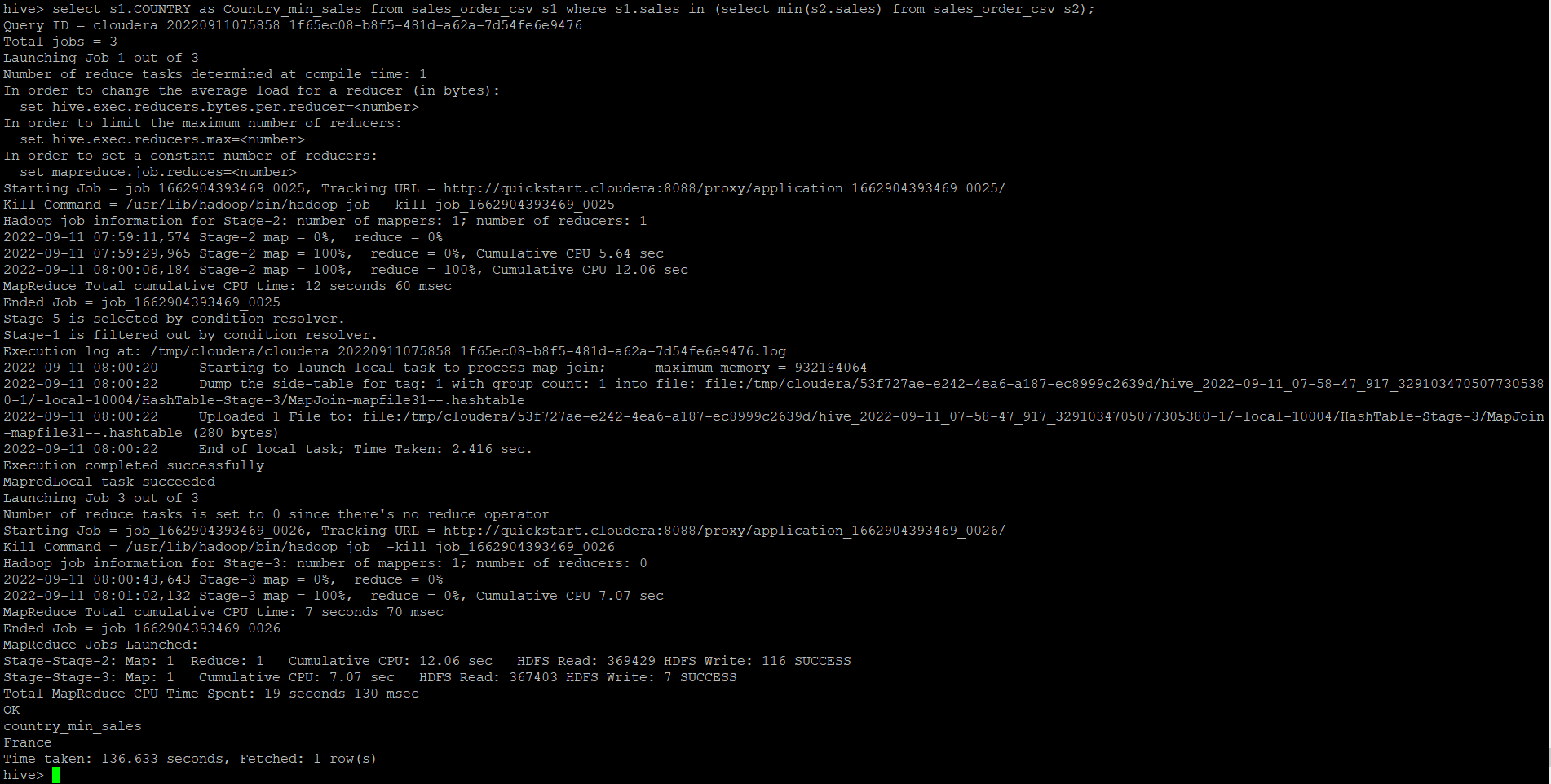


**e. In which country sales was maximum and in which country sales was minimum**

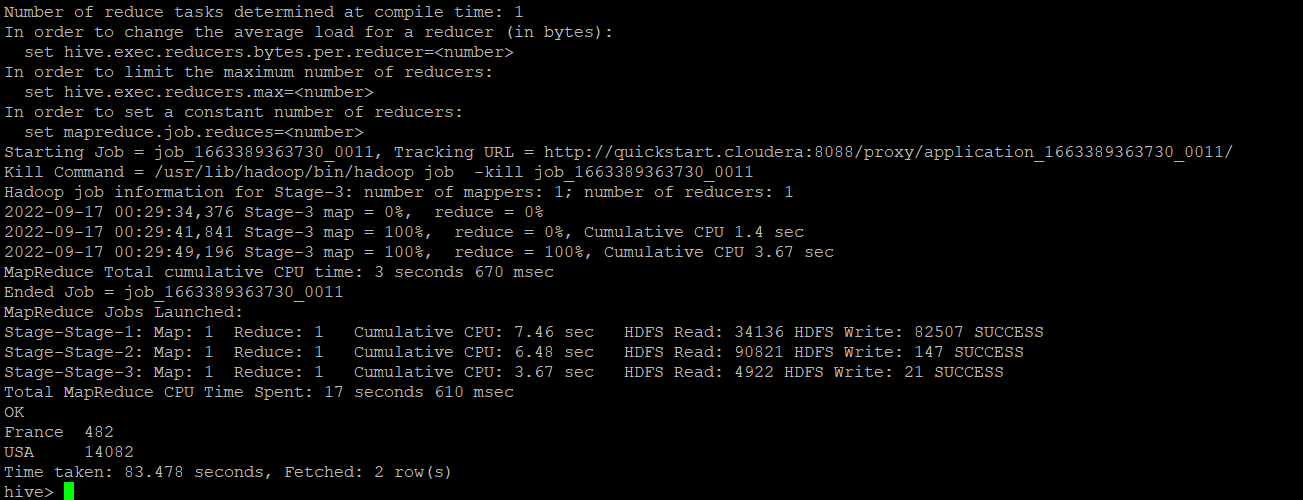
**Ans:- hive> select s1.COUNTRY as Country\_max\_sales from sales\_order\_csv s1 where s1.sales in (select max(s2.sales) from sales\_order\_csv s2);**

**hive> select s1.COUNTRY as Country\_min\_sales from sales\_order\_csv s1 where s1.sales in (select min(s2.sales) from sales\_order\_csv s2);**

****

****

**Hive> select A.country,A.sales from (select country,sales,rank() over (order by sales) rnk\_min,rank() over(order by sales desc) rnk\_maxfrom sales\_order\_orc)A where rnk\_min = 1 or rnk\_max =1 order by sales;**

****

**f. Calculate quarterly sales for each city**

Ans:- **hive>select quarterly\_sales,city from (select sum(s.sales) as quarterly\_sales,s.QTR\_ID,s.city from sales\_order\_csv s group by s.QTR\_ID,s.city)a;**

**quarterly\_sales city**

**56181.320068359375 Bergamo**

**31606.72021484375 Boras**

**31474.7802734375 Brickhaven**

**16118.479858398438 Brisbane**

**18800.089721679688 Bruxelles**

**37850.07958984375 Burbank**

**13529.570190429688 Burlingame**

**21782.699951171875 Cambridge**

**16628.16015625 Charleroi**

**26906.68017578125 Cowes**

**38784.470458984375 Dublin**

**51373.49072265625 Espoo**

**48698.82922363281 Frankfurt**

**50432.549560546875 Gensve**

**3987.199951171875 Glendale**

**8775.159912109375 Graz**

**26422.819458007812 Helsinki**

**58871.110107421875 Kobenhavn**

**20178.1298828125 Lille**

**8477.219970703125 London**

**23889.320068359375 Los Angeles**

**9748.999755859375 Lule**

**101339.13977050781 Lyon**

**357668.4899291992 Madrid**

**55245.02014160156 Makati City**

**51017.919860839844 Manchester**

**2317.43994140625 Marseille**

**49637.57067871094 Melbourne**

**38191.38977050781 Minato-ku**

**32647.809814453125 NYC**

**59617.39978027344 Nantes**

**12133.25 Nashua**

**48578.95935058594 New Bedford**

**8722.1201171875 Newark**

**65012.41955566406 North Sydney**

**50490.64013671875 Osaka**

**49055.40026855469 Oulu**

**71494.17944335938 Paris**

**44273.359436035156 Pasadena**

**27398.820434570312 Philadelphia**

**52029.07043457031 Reims**

**87489.23010253906 San Diego**

**72899.19995117188 San Francisco**

**267315.2586669922 San Rafael**

**28395.18994140625 Singapore**

**21730.029907226562 South Brisbane**

**54701.999755859375 Stavern**

**15139.1201171875 Toulouse**

**5759.419921875 Versailles**

**6166.7998046875 Allentown**

**4219.2001953125 Barcelona**

**74994.240234375 Boston**

**7277.35009765625 Brickhaven**

**75778.99060058594 Bridgewater**

**8411.949829101562 Bruxelles**

**14380.920043945312 Cambridge**

**1711.260009765625 Charleroi**

**43971.429931640625 Chatswood**

**31018.230102539062 Espoo**

**14378.089965820312 Glen Waverly**

**20350.949768066406 Glendale**

**62091.880615234375 Kobenhavn**

**33847.61975097656 Las Vegas**

**91211.0595703125 Liverpool**

**32376.29052734375 London**

**339588.0513305664 Madrid**

**52481.840087890625 Marseille**

**60135.84033203125 Melbourne**

**26482.700256347656 Minato-ku**

**58257.50012207031 Montreal**

**165100.33947753906 NYC**

**60344.990173339844 Nantes**

**36973.309814453125 New Haven**

**74506.06909179688 Newark**

**17114.43017578125 Osaka**

**17813.40008544922 Oulu**

**80215.4203491211 Paris**

**7287.240234375 Philadelphia**

**41509.94006347656 Reggio Emilia**

**18971.959716796875 Reims**

**98104.24005126953 Salzburg**

**160010.27026367188 San Jose**

**7261.75 San Rafael**

**92033.77014160156 Singapore**

**80438.47985839844 Strasbourg**

**31302.500244140625 Tsawassen**

**71930.61041259766 Allentown**

**16363.099975585938 Bergen**

**53941.68981933594 Boras**

**15344.640014648438 Boston**

**114974.53967285156 Brickhaven**

**34100.030029296875 Brisbane**

**47760.479736328125 Bruxelles**

**42031.83020019531 Burlingame**

**48828.71942138672 Cambridge**

**1637.199951171875 Charleroi**

**69694.40002441406 Chatswood**

**18971.959838867188 Dublin**

**31569.430053710938 Espoo**

**67281.00903320312 Gensve**

**12334.819580078125 Glen Waverly**

**7600.1201171875 Glendale**

**42744.0595703125 Helsinki**

**34453.84973144531 Las Vegas**

**69714.09008789062 Madrid**

**34993.92004394531 Munich**

**63027.92004394531 NYC**

**61310.880126953125 Nantes**

**45738.38952636719 New Bedford**

**47191.76013183594 North Sydney**

**34145.47021484375 Oslo**

**37501.580322265625 Oulu**

**27798.480102539062 Paris**

**55776.119873046875 Pasadena**

**56421.650390625 Reggio Emilia**

**15146.31982421875 Reims**

**6693.2802734375 Salzburg**

**216297.40063476562 San Rafael**

**90250.07995605469 Singapore**

**10640.290161132812 South Brisbane**

**94117.25988769531 Torino**

**17251.08056640625 Toulouse**

**43332.349609375 Tsawassen**

**100595.5498046875 Aaarhus**

**44040.729736328125 Allentown**

**74192.66003417969 Barcelona**

**81774.40008544922 Bergamo**

**95277.17993164062 Bergen**

**48710.92053222656 Boras**

**63730.7802734375 Boston**

**11528.52978515625 Brickhaven**

**26115.800537109375 Bridgewater**

**8234.559936523438 Burbank**

**65221.67004394531 Burlingame**

**54251.659912109375 Cambridge**

**13463.480224609375 Charleroi**

**37905.14990234375 Chatswood**

**51334.15966796875 Cowes**

**36472.76025390625 Frankfurt**

**37878.54992675781 Glen Waverly**

**34485.49987792969 Glendale**

**43488.740234375 Graz**

**42083.499755859375 Helsinki**

**24078.610107421875 Kobenhavn**

**100306.58020019531 Koln**

**14449.609741210938 Las Vegas**

**48874.28088378906 Lille**

**26797.210083007812 Liverpool**

**83970.029296875 London**

**24159.14013671875 Los Angeles**

**66005.8798828125 Lule**

**41535.11022949219 Lyon**

**315580.80963134766 Madrid**

**38770.71032714844 Makati City**

**106789.88977050781 Manchester**

**20136.859985351562 Marseille**

**91221.99914550781 Melbourne**

**55888.65026855469 Minato-ku**

**15947.290405273438 Montreal**

**300011.6999511719 NYC**

**23031.589599609375 Nantes**

**119552.04949951172 Nashua**

**113557.509765625 New Bedford**

**42498.760498046875 New Haven**

**41791.949462890625 North Sydney**

**45078.759765625 Oslo**

**89436.60034179688 Paris**

**4512.47998046875 Pasadena**

**116503.07043457031 Philadelphia**

**44669.740478515625 Reggio Emilia**

**48895.59014892578 Reims**

**45001.10986328125 Salzburg**

**151459.4805908203 San Francisco**

**163983.64880371094 San Rafael**

**54723.621154785156 Sevilla**

**77809.37023925781 Singapore**

**27098.800048828125 South Brisbane**

**61897.19006347656 Stavern**

**38098.240234375 Toulouse**

**75238.91955566406 Vancouver**

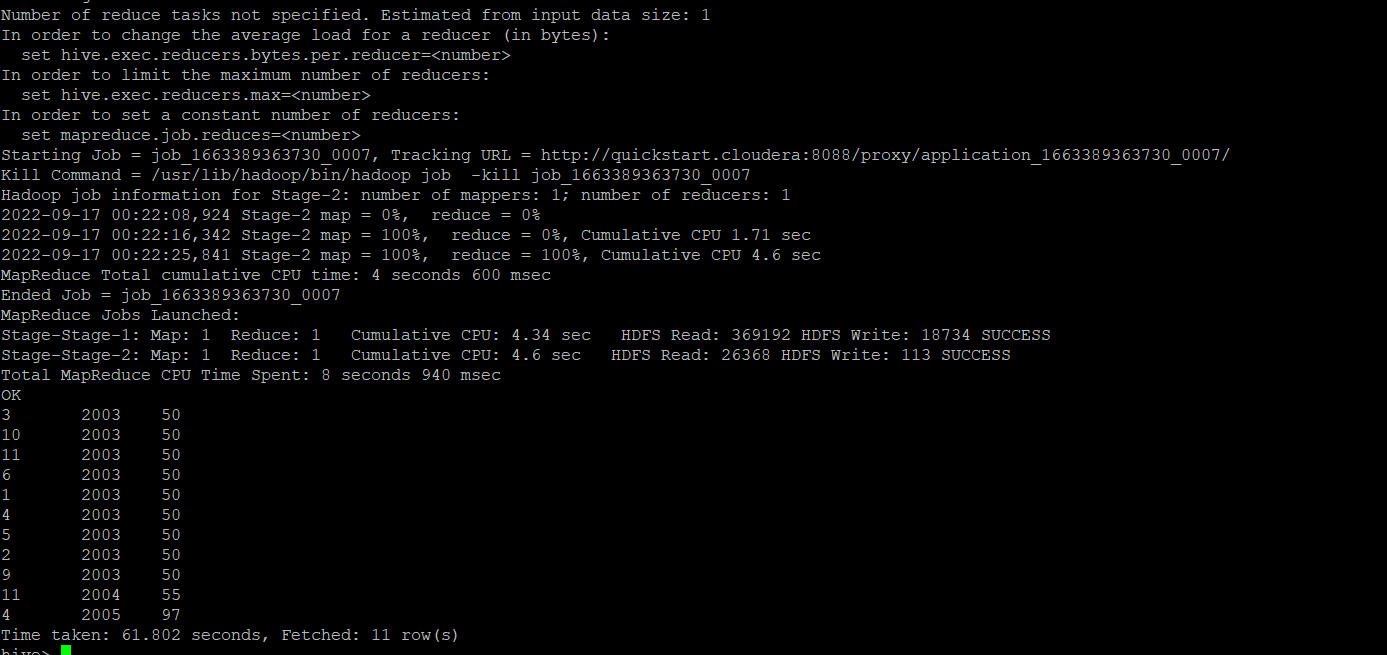
**59074.90026855469 Versailles**

**85555.98962402344 White Plains**

**Time taken: 24.484 seconds, Fetched: 182 row(s)**

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

**Ans:- Hive> select month\_id,year\_id,QUANTITYORDERED from (select distinct 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;**

****