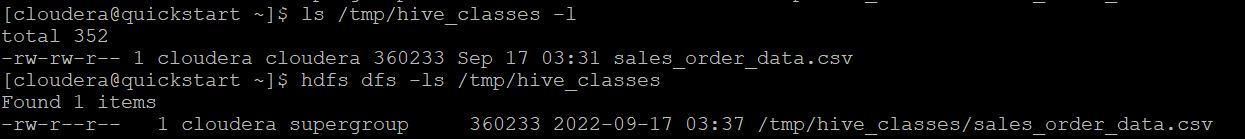
1. Downloaded Data
2. Store raw data into hdfs location

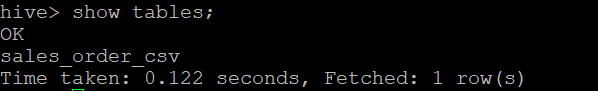
ls /tmp/hive\_classes -l -> On Local Path

hdfs dfs -ls /tmp/hive\_classes -> HDFS Location



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\_csv(ORDERNUMBER int, QUANTITYORDERED int, PRICEEACH float, ORDERLINENUMBER int,SALES float, STATUS string, QTR\_ID int, MONTH\_ID int, YEAR\_ID int, PRODUCTCODE int, MSRP string, PHONE int, CITY string, STATE string, POSTALCODE int, COUNTRY string, TERRITORY string, CONTACTLASTNAME string, CONTACTFIRSTNAME string, DEALSIZE string) row format delimited fields terminated by ',' lines terminated by '\n' tblproperties ("skip.header.line.count"="1");



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



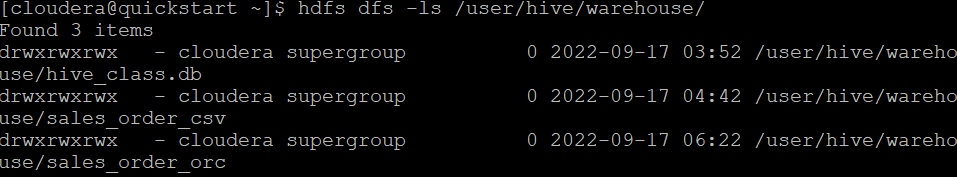
load data inpath '/tmp/hive\_classes1/sales\_order\_data.csv' into table sales\_order\_csv;

set hive.cli.print.header = true;

select \* from sales\_order\_csv limit 10;

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

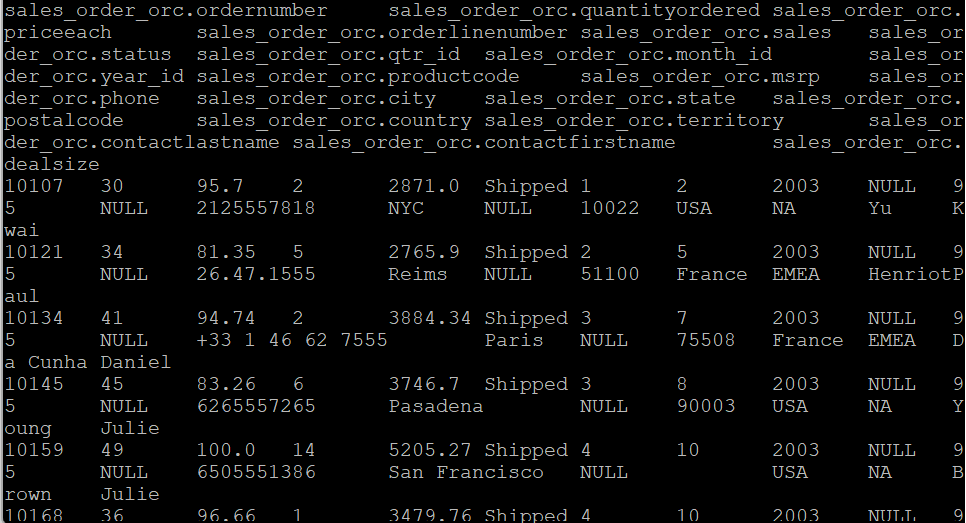
create table sales\_order\_csv(ORDERNUMBER int, QUANTITYORDERED int, PRICEEACH float, ORDERLINENUMBER int,SALES float, STATUS string, QTR\_ID int, MONTH\_ID int, YEAR\_ID int, PRODUCTCODE int, MSRP string, PHONE int, CITY string, STATE string, POSTALCODE int, COUNTRY string, TERRITORY string, CONTACTLASTNAME string, CONTACTFIRSTNAME string, DEALSIZE string) stored as orc;



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

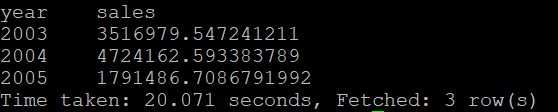
from sales\_order\_csv insert overwrite table sales\_order\_orc select \*;

set hive.cli.print.header=true;



1. Calculatye total sales per year

select YEAR\_ID as Year, sum(SALES) as Sales from sales\_order\_csv group by YEAR\_ID;

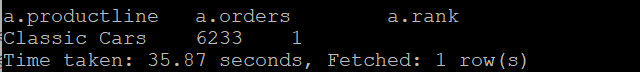


1. Find a product for which maximum orders were placed

select productline, sum(ORDERLINENUMBER) as orders from sales\_order\_orc group by productline order by orders desc;

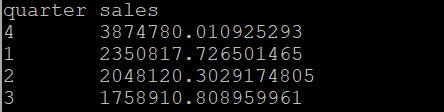
or

select a.\* from(select productline, sum(ORDERLINENUMBER) as orders, dense\_rank() over(order by sum(orderlinenumber) desc) as rank from sales\_order\_orc group by productline) a where rank=1;



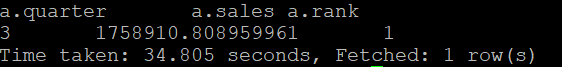
1. Calculate the total sales for each quarter

select QTR\_ID as quarter, sum(sales) as sales from sales\_order\_orc group by QTR\_ID order by sales desc;



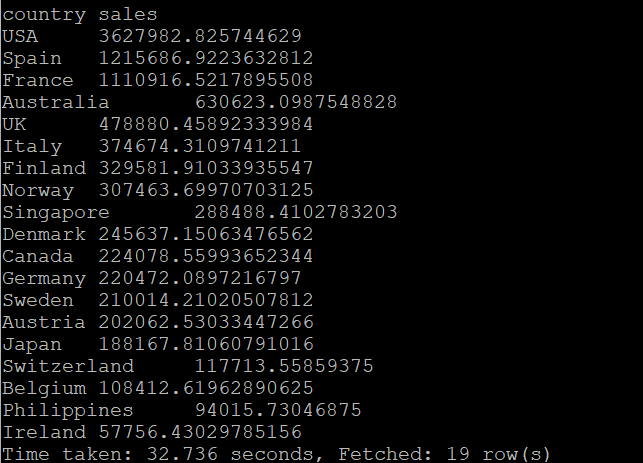
1. In which quarter sales was minimum

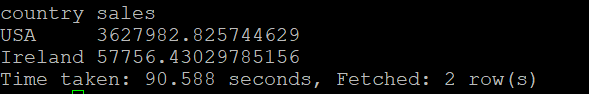
select a.\* from(select QTR\_ID as quarter, sum(sales) as sales, dense\_rank() over(order by sum(sales) asc) as rank from sales\_order\_orc group by QTR\_ID) a where a.rank=1;



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

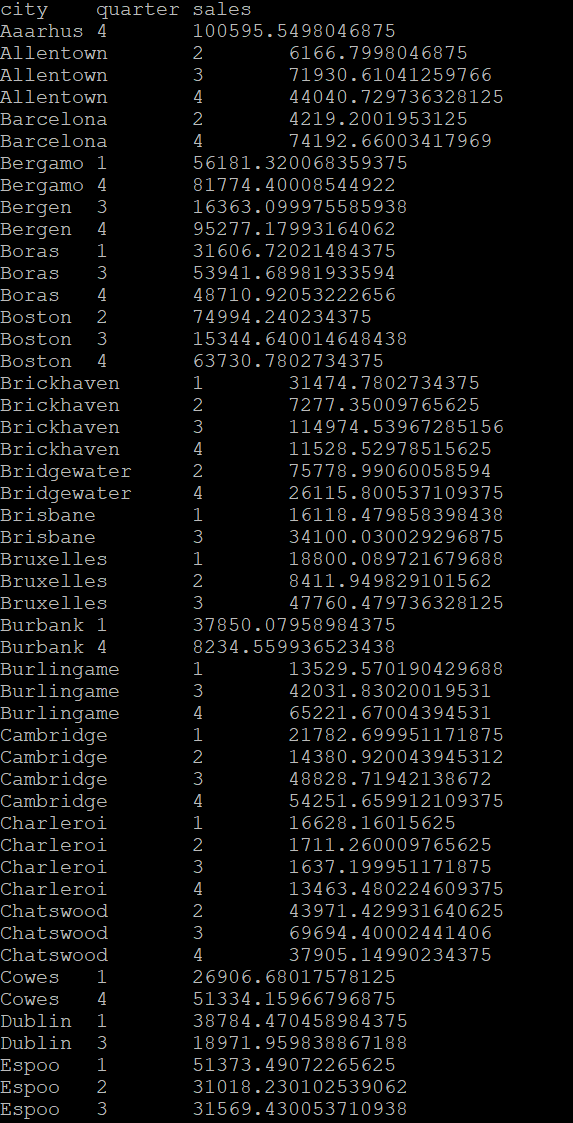
select b.country as country, b.sales as sales from(select a.country as country, a.sales as sales from(select country, sum(sales) as sales, dense\_rank() over(order by sum(sales) asc) as rank from sales\_order\_orc group by country) a where a.rank=1 union all select a.country as country,a.sales as sales from(select country, sum(sales) as sales, dense\_rank() over(order by sum(sales) desc) as rank from sales\_order\_orc group by country) a where a.rank=1) b order by sales desc;





1. Calculate quartelry sales for each city

select city,qtr\_id as quarter, sum(sales) as 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 a.year\_number as year, a.month\_number as month, a.quantities\_sold as quantities\_sold from(select year\_id as year\_number, month\_id as month\_number,sum(QUANTITYORDERED) as quantities\_sold, dense\_rank() over(partition by year\_id order by sum(QUANTITYORDERED) desc) as rank from sales\_order\_orc group by year\_id, month\_id) a where a.rank=1;

