2. Store raw data into hdfs location

cd filepath

hadoop fs -put filepath/sales.csv /user

hadoop fs -ls /user

haddop fs -cat /user/sales.csv

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

create table sales\_order\_csv if not exists sales\_order\_csv

(ORDERNUMBER int , QUANTITYORDERED int,PRICEEACH int,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 ',' lines terminated by '\n'

STORED AS TEXTFILE LOCATION '/storage/sales.csv' TBLPROPERTIES('skip.header.line.count'='1');`

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

LOAD DATA INPATH '/user/sales\_order\_csv' INTO TABLE sales\_order\_orc

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

Create table sales\_order\_orc

( ORDER NUMBER int , QUANTITY ORDERED int,PRICEEACH int,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,POSTAL CODE string,COUNTRY string,TERRITORY string,CONTACT LASTNAME string,

CONTACT FIRSTNAME string,DEALSIZE string)

STORED AS ORC;

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

INSERT INTO sales\_order\_orc SELECT \* FROM sales\_order\_orc

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

a. Calculate total sales per year

Ans. SELECT year\_id AS year, SUM(sales) AS total\_sales

FROM sales

GROUP BY year\_id;

b. Find a product for which maximum orders were placed

Ans. SELECT productline, SUM(quantityordered) AS total\_quantity

FROM sales

GROUP BY productline

ORDER BY total\_quantity DESC

LIMIT 1;

c. Calculate the total sales for each quarter

Ans. SELECT QTR\_ID as Qtr , SUM(sales) as total\_sales\_per\_quater

FROM sales

GROUP BY QTR\_ID;

d. In which quarter sales was minimum \*

Ans. SELECT QTR\_ID as Qtr , min(sales) as total\_sales\_per\_quater

FROM sales

GROUP BY QTR\_ID

LIMIT 1;

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

Ans. SELECT country, sum(sales) as total\_sales\_per\_country

FROM sales

GROUP BY country

ORDER BY total\_sales\_per\_country DESC

LIMIT 1;

SELECT country, sum(sales) as total\_sales\_per\_country

FROM sales

GROUP BY country

ORDER BY total\_sales\_per\_country

limit 1;

f. Calculate quartelry sales for each city

Ans. SELECT city, qtr\_id as quarter, SUM(sales) as total\_sales\_per\_quarter\_per\_city

FROM sales

GROUP BY city,qtr\_id;

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

Ans.

with temp as

(SELECT year\_id as year, month\_id as month,

SUM(quantityordered) as total\_quantity\_per\_month

FROM sales

GROUP BY year\_id, month\_id

ORDER BY year\_id,total\_quantity\_per\_month desc),

temp2 as (

select year,month,total\_quantity\_per\_month,

rank() over (partition by year order by

total\_quantity\_per\_month desc) rnk

from temp )

select year,month,total\_quantity\_per\_month from temp2

where rnk=1