1) Downloaded the zip file from git. Placed the whole folder into the cloudera server using filezill from local machine.

2) Copied the 'sales_order_csv' data from linux machine to hdfs by the following command:

=> hadoop fs -copyFromLocal /home/cloudera/vishwa_hdfs/Hive-Class-main /user/practice/

3) Created the "sales_order_csv" table in hive.

```
=> 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 ","
   > stored as textfile tblproperties ("skip.header.line.count"="1");
```

Above command returned OK.

4) Load sales_order_data.csv from hdfs location to table sales_order_csv

```
=>  load data inpath '/user/practice/Hive-Class-main/sales_order_data.csv' into table sales_order_csv;
```

5) Create orc table

```
=> 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;
```

6) Load data from sales_order_csv to sales_order_orc

=> insert into sales_order_orc select * from sales_order_csv;

CALCULATIONS

a) calculate total sales per year:

=> select year_id, sum(sales) as total_sales from sales_order_orc group by year_id;

=> output:

year_id total_sales

2003 3516979.547241211

2004 4724162.593383789

2005 1791486.7086791992

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

logic: Every row has a unique order_id and each order_id has ordered a product having the productcode so if we count the duplicates in column PRODUCTCODE we get the duplicates >= 1 then we can order it in descending order and get first row.

=> select productcode, count(productcode) as maximum_orders from sales_order_csv group by productcode having count(productcode) >= 1 order by maximum_orders desc limit 1;

=> output:

productcode maximum_orders

S18_3232 52

c) calculate total sales from each quarter.

=> select qtr_id, sum(sales) as total_sales from sales_order_orc group by qtr_id order by total_sales limit 5;

=> output:

qtr_id total_sales

3 1758910.808959961

2 2048120.3029174805

1 2350817.726501465

4 3874780.010925293

d) 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 desc limit 1;

=> output:

qtr_id	total_sales
4	3874780.010925293

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

=> with new_table as (select country, sum(sales) as total_sales from sales_order_orc group by country) select country, total_sales from new_table order by total_sales asc limit 1 union all select country, total_sales from new_table order by total_sales desc limit 1;

=> output:

_u1.country	_u1.total_sales
Ireland	57756.43029785156
USA	3627982.825744629

f) calculate quarterly sales for each city

=> select city, qtr_id, sum(sales) from sales_order_orc group by qtr_id, city order by city limit 10;

=> output:

city	qtr_id	_c2
Aaarhus	4	100595.5498046875
Allentown	4	44040.729736328125
Allentown	2	6166.7998046875
Allentown	3	71930.61041259766
Barcelona	4	74192.66003417969
Barcelona	2	4219.2001953125
Bergamo	4	81774.40008544922
Bergamo	1	56181.320068359375
Bergen	3	16363.099975585938
Bergen	4	95277.17993164062

g) Find a month for each year in which maximum number of quantities whre sold

=> select year_id, month_id, total_orders from (select *, rank() over (partition by year_id order by total_orders desc) rnk from (select year_id, month_id, sum(quantityordered) total_orders from sales_order_orc group by year_id, month_id order by year_id, month_id) a) b where rnk=1;

=> output:year_id month_id total_orders

2003 11 10179

2004 11 10678

2005 5 4357