**1. Table creation and loading data into hive warehouse.**

CREATE TABLE ecom\_data (

order\_id STRING,

customer\_id STRING,

quantity INT,

price\_MRP FLOAT,

payment FLOAT,

timestamp STRING,

rating INT,

product\_category STRING,

product\_id STRING,

payment\_type STRING,

order\_status STRING,

product\_weight\_g INT,

product\_length\_cm INT,

product\_height\_cm INT,

product\_width\_cm INT,

customer\_city STRING,

customer\_state STRING,

seller\_id STRING,

seller\_city STRING,

seller\_state STRING,

payment\_installments INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

TBLPROPERTIES("skip.header.line.count"="1");

**loading data into table;**

load data inpath 'ecom\_hive\_project/ecom\_data' into table ecom\_data;

**creating new empty table to remove timestamp issues and duplicate values**

CREATE TABLE ecom(

order\_id STRING,

customer\_id STRING,

quantity INT,

price\_MRP FLOAT,

payment FLOAT,

timestamp timestamp,

rating INT,

product\_category STRING,

product\_id STRING,

payment\_type STRING,

order\_status STRING,

product\_weight\_g INT,

product\_length\_cm INT,

product\_height\_cm INT,

product\_width\_cm INT,

customer\_city STRING,

customer\_state STRING,

seller\_id STRING,

seller\_city STRING,

seller\_state STRING,

payment\_installments INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

TBLPROPERTIES("skip.header.line.count"="1");

**Inserting value in newly created table**

INSERT OVERWRITE TABLE ecom

SELECT DISTINCT order\_id,

customer\_id,

quantity,

price\_MRP,

payment,

CASE

WHEN timestamp LIKE '%-%' THEN from\_unixtime(unix\_timestamp(timestamp, 'dd-MM-yyyy HH:mm'))

WHEN timestamp LIKE '%/%' THEN from\_unixtime(unix\_timestamp(timestamp, 'dd/MM/yyyy hh:mm'))

ELSE NULL

END,

rating,

product\_category,

product\_id,

payment\_type,

order\_status,

product\_weight\_g,

product\_length\_cm,

product\_height\_cm,

product\_width\_cm,

customer\_city,

customer\_state,

seller\_id,

seller\_city,

seller\_state,

payment\_installments

FROM (

SELECT \*,

dense\_rank() OVER (PARTITION BY order\_id, product\_id ORDER BY quantity DESC) AS `ranking`

FROM ecom\_data

) t

WHERE ranking = 1;

**-> HIVE JOBS................**

1. **Customer Segmentation**

**Categorizing customers based on their spendings**

SELECT customer\_id,SUM(payment) AS spending, CASE WHEN SUM(payment) < 25000 THEN 'Under Affluent' WHEN SUM(payment) BETWEEN 25000 AND 75000 THEN 'Semi-affluent' WHEN SUM(payment) > 75000 THEN 'Affluent' END AS Categories FROM ecom\_data GROUP BY customer\_id ORDER BY spending DESC;

**-> creating external table**

create external table customer\_segmentation(Customer\_id string, Spending float, Categories string) row format delimited fields terminated by ' ' location '/user/hive/warehouse/problem\_1/result.txt';

**->loading data into external table**

insert overwrite table customer\_segmentation SELECT customer\_id,SUM(payment) AS spending, CASE WHEN SUM(payment) < 25000 THEN 'Under-Affluent' WHEN SUM(payment) BETWEEN 25000 AND 75000 THEN 'Semi-affluent' WHEN SUM(payment) > 75000 THEN 'Affluent' END AS Categories FROM ecom\_data GROUP BY customer\_id ORDER BY spending DESC;

**-> create empty table in database**

create table customer\_segmentation(Customer\_id varchar(255), Spending float, Categories varchar(30));

**-> exporting data from hive to database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table customer\_segmentation --export-dir /user/hive/warehouse/problem\_1/result.txt/000000\_0 --input-fields-terminated-by ' '

**2. Monthly Trend Forecasting**

**The monthly trend of sales**

with cte as (select year(timestamp) as year, month(timestamp) as month, product\_category, sum(quantity) as count, rank() over(partition by year(timestamp), month(timestamp) order by sum(quantity) desc) as r, sum(payment) as spend from ecom group by year(timestamp), month(timestamp), product\_category) select cte.year, cte.month, cte.product\_category, cte.count, spend from cte where r=1 order by year desc, spend desc;

**-> create external table**

create external table monthly\_trend(year int, month int, product string, total\_no\_product int, total\_sales float) row format delimited fields terminated by ' ' location '/user/hive/warehouse/program\_2/result.txt';

**-> load data into external table**

with cte as (select year(timestamp) as year, month(timestamp) as month, product\_category, sum(quantity) as count, rank() over(partition by year(timestamp), month(timestamp) order by sum(quantity) desc) as r, sum(payment) as spend from ecom group by year(timestamp), month(timestamp), product\_category) insert overwrite table monthly\_trend select cte.year, cte.month, cte.product\_category, cte.count, spend from cte where r=1 order by year desc, spend desc;

**-> creating empty table in database**

create table monthly\_trend(year int, month int, product varchar(100), total\_no\_product int, total\_sales float);

**-> export data to empty table**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table monthly\_trend --export-dir /user/hive/warehouse/program\_2/result.txt/000000\_0 --input-fields-terminated-by ' '

**3. Hourly Sales Analysis**

**Which hour has more no. of sales?**

select hour(timestamp) as hour, sum(quantity) as no\_of\_sale from ecom group by hour(timestamp) order by no\_of\_sale desc limit 1;

**-> create external table**

create external table hourly\_sales(hour int, no\_of\_sales int) row format delimited fields terminated by ' ' location '/user/hive/warehouse/problem\_3/result.txt';

**-> loading data into external table**

insert overwrite table hourly\_sales select hour(timestamp) as hour, sum(quantity) as no\_of\_sale from ecom group by hour(timestamp) order by no\_of\_sale desc limit 1;

**-> create empty table in database**

create table hourly\_sales(hour int, no\_of\_sales int);

**-> export data in empty table**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table hourly\_sales --export-dir /user/hive/warehouse/problem\_3/result.txt/000000\_0 --input-fields-terminated-by ' '

**4. Product Based Analysis**

**Which category product has sold more?**

select product\_category, sum(quantity) as count from ecom group by product\_category order by count desc limit 1;

**-> create external table**

create external table product\_1(product string, count int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_1/result.txt'

**-> load data in external table**

insert overwrite table product\_1 select product\_category, sum(quantity) as count from ecom group by product\_category order by count desc limit 1;

**->create empty table**

create table product\_sold(product varchar(100), count int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table product\_sold --export-dir /user/hive/wareshouse/problem\_4\_1/result.txt/000000\_0 --input-fields-terminated-by ' '

**Which category product has more rating?**

select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating desc limit 1;

**-> create external table**

create external table product\_2(product string, avg\_rating float) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_2/result.txt'

**-> load data in external table**

insert overwrite table product\_2 select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating desc limit 1;

**->create empty table**

create table product\_rating(product varchar(100), rating float);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table product\_rating --export-dir /user/hive/wareshouse/problem\_4\_2/result.txt/000000\_0 --input-fields-terminated-by ' '

**Which product has sold more?**

select product\_category, sum(quantity) as no\_of\_sold from ecom group by product\_category order by no\_of\_sold desc limit 1;

**-> create external table**

create external table product\_3(product string, no\_of\_sold int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_3/result.txt';

**-> load data in external table**

insert overwrite table product\_3 select product\_category, sum(quantity) as no\_of\_sold from ecom group by product\_category order by no\_of\_sold desc limit 1;

**->create empty table**

create table highest\_product\_sold(product varchar(100), no\_of\_sold int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table highest\_product\_sold --export-dir /user/hive/wareshouse/problem\_4\_3/result.txt/000000\_0 --input-fields-terminated-by ' '

**Top 10 highest & least product rating?**

**Highest rating**

select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating desc limit 10;

**-> create external table**

create external table product\_4i(product string, avg\_rating float) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_4i/result.txt';

**-> load data in external table**

insert overwrite table product\_4i select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating desc limit 10;

**->create empty table**

create table top\_product\_rating(product varchar(100), rating float);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table top\_product\_rating --export-dir /user/hive/wareshouse/problem\_4\_4i/result.txt/000000\_0 --input-fields-terminated-by ' '

**--------------least rating---------------------**

select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating asc limit 10;

**-> create external table**

create external table product\_4ii(product string, avg\_rating float) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_4ii/result.txt';

**-> load data in external table**

insert overwrite table product\_4ii select product\_category, avg(rating) as avg\_rating from ecom group by product\_category order by avg\_rating asc limit 10;

**->create empty table**

create table least\_product\_rating(product varchar(100), rating float);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table least\_product\_rating --export-dir /user/hive/wareshouse/problem\_4\_4ii/result.txt/000000\_0 --input-fields-terminated-by ' '

**Order Count for each rating**

select rating, count(rating)as count from ecom group by rating order by count desc;

**-> create external table**

create external table product\_5(product string, avg\_rating int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_4\_5/result.txt';

**-> load data in external table**

insert overwrite table product\_5 select rating, count(rating)as count from ecom group by rating order by count desc;

**->create empty table**

create table count\_of\_rating(rating int, count\_of\_rating int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table count\_of\_rating --export-dir /user/hive/wareshouse/problem\_4\_5/result.txt/000000\_0 --input-fields-terminated-by ' '

**5. Payment Preference**

**What are the most commonly used payment types?**

select payment\_type, count(payment\_type) as no\_of\_times from ecom group by payment\_type order by no\_of\_times desc;

**-> create external table**

create external table payment\_1(payment\_type string, no\_of\_times int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_5/result.txt';

**-> load data in external table**

insert overwrite table payment\_1 select payment\_type, count(payment\_type) as no\_of\_times from ecom group by payment\_type order by no\_of\_times desc;

**->create empty table**

create table count\_of\_payment\_mode(payment\_type varchar(30), no\_of\_times int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table count\_of\_payment\_mode --export-dir /user/hive/wareshouse/problem\_5/result.txt/000000\_0 --input-fields-terminated-by ' '

**Count of Orders With each No. of Payment Installments**

select payment\_installments, count(\*) as count\_orders from ecom where payment\_installments is not null group by payment\_installments;

**-> create external table**

create external table payment\_2(payment\_installments int, count\_orders int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_5ii/result.txt';

**-> load data in external table**

insert overwrite table payment\_2 select payment\_installments, count(\*) as count\_orders from ecom where payment\_installments is not null group by payment\_installments;

**->create empty table**

create table Count\_of\_Orders(payment\_installments int, count\_of\_orders int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table Count\_of\_Orders --export-dir /user/hive/wareshouse/problem\_5ii/result.txt/000000\_0 --input-fields-terminated-by ' '

**6. Potential Customer's Location**

**Where do most customers come from?**

select customer\_state, count(customer\_state) as no\_of\_customer from ecom group by customer\_state order by no\_of\_customer desc;

**-> create external table**

create external table location(customer\_state string, no\_of\_customer int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_6/result.txt';

**-> load data in external table**

insert overwrite table location select customer\_state, count(customer\_state) as no\_of\_customer from ecom group by customer\_state order by no\_of\_customer desc;

**->create empty table**

create table Customer\_location(customer\_state varchar(30), no\_of\_customer int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table Customer\_location --export-dir /user/hive/wareshouse/problem\_6/result.txt/000000\_0 --input-fields-terminated-by ' '

**7. Seller Rating**

**Which seller sold more?**

select seller\_id, count(quantity) as sold from ecom group by seller\_id order by sold desc limit 1;

**-> create external table**

create external table seller\_sold(seller\_id string, sold int) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_7\_1/result.txt';

**-> load data in external table**

insert overwrite table seller\_sold select seller\_id, count(quantity) as sold from ecom group by seller\_id order by sold desc limit 1;

**->create empty table**

create table seller\_sold(seller\_id varchar(200), sold int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table seller\_sold --export-dir /user/hive/wareshouse/problem\_7\_1/result.txt/000000\_0 --input-fields-terminated-by ' '

**Which seller got more rating?**

SELECT seller\_id, rating FROM (SELECT seller\_id, AVG(rating) AS rating, RANK() OVER (ORDER BY AVG(rating) DESC) AS ranking FROM ecom GROUP BY seller\_id) temp WHERE ranking = 1 ORDER BY rating DESC;

**-> create external table**

create external table seller\_rating(seller\_id string, rating float) row format delimited fields terminated by ' ' location '/user/hive/wareshouse/problem\_7\_2/result.txt';

**-> load data in external table**

insert overwrite table seller\_rating SELECT seller\_id, rating FROM (SELECT seller\_id, AVG(rating) AS rating, RANK() OVER (ORDER BY AVG(rating) DESC) AS ranking FROM ecom GROUP BY seller\_id) temp WHERE ranking = 1 ORDER BY rating DESC;

**->create empty table**

create table seller\_rating\_higher(seller\_id varchar(200), rating float);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table seller\_rating\_higher --export-dir /user/hive/wareshouse/problem\_7\_2/result.txt/000000\_0 --input-fields-terminated-by ' '

**8. Logistics based Optimization Insights**

**Which city buys heavy weight products and low weight products?**

**Higher Weight**

select customer\_city, sum(product\_weight\_g) as weight from ecom group by customer\_city order by weight desc limit 10; -- high weight product

**-> create external table**

create external table higher\_weight(customer\_city string, weight int) row format delimited fields terminated by ',' location '/user/hive/wareshouse/problem\_8/result.txt';

**-> load data in external table**

insert overwrite table higher\_weight select customer\_city, sum(product\_weight\_g) as weight from ecom group by customer\_city order by weight desc limit 10;

**->create empty table**

create table higher\_weight(customer\_city varchar(20), weight int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table higher\_weight --export-dir /user/hive/wareshouse/problem\_8/result.txt/000000\_0 --input-fields-terminated-by ','

**-----------------lower weight----------------**

select customer\_city, sum(product\_weight\_g) as weight from ecom group by customer\_city order by weight asc limit 10; -- low weight product

**-> create external table**

create external table lower\_weight(customer\_city string, weight int) row format delimited fields terminated by ',' location '/user/hive/wareshouse/problem\_8\_1/result.txt';

**-> load data in external table**

insert overwrite table lower\_weight select customer\_city, sum(product\_weight\_g) as weight from ecom group by customer\_city order by weight asc limit 10;

**->create empty table**

create table low\_weight(customer\_city varchar(20), weight int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table low\_weight --export-dir /user/hive/wareshouse/problem\_8\_1/result.txt/000000\_0 --input-fields-terminated-by ','

**How much products sold within seller state?**

select sum(quantity) as count from ecom where seller\_state = customer\_state;

**-> create external table**

create external table product\_sold(quantity int) row format delimited fields terminated by ',' location '/user/hive/wareshouse/problem\_8\_2/result.txt';

**-> load data in external table**

insert overwrite table product\_sold select sum(quantity) as count from ecom where seller\_state = customer\_state;

**->create empty table**

create table product\_sold(No\_of\_product\_sold\_within\_seller\_state int);

**->export data to the database**

sqoop export --connect jdbc:mysql://localhost:3306/ecom --username root --password cloudera --table product\_sold\_within\_seller\_state --export-dir /user/hive/wareshouse/problem\_8\_2/result.txt/000000\_0 --input-fields-terminated-by ','