HIVE HEALTHCARE PROJECT

Problem Statement 1: Jimmy, from the healthcare department, has requested a report that shows how the number of treatments each age category of patients has gone through in the year 2022. The age category is as follows, Children (00-14 years), Youth (15-24 years), Adults (25-64 years), and Seniors (65 years and over). Assist Jimmy in generating the report.

```
hive> SELECT COUNT(*), v1.category
    > FROM (
    > SELECT
    > CASE
    > WHEN YEAR(t1.date) - YEAR(dob) <= 14 THEN 'children'
    > WHEN YEAR(t1.date) - YEAR(dob) <= 24 THEN 'youth'
    > WHEN YEAR(t1.date) - YEAR(dob) <= 64 THEN 'adults'
    > ELSE 'senior citizen'
    > END AS category,
    > p.patientid AS patientid
    > FROM Patient p
    > INNER JOIN treatment t1 ON p.patientid = t1.patientid
    > WHERE YEAR(t1.date) = 2022
    > ) AS v1
    > GROUP BY v1.category;

1404    adults
788     children
699     senior citizen
76      youth
Time taken: 74.184 seconds, Fetched: 4 row(s)
```

Creating External Table:

```
hive> create external table s1_p1(count int, category varchar(50))
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > location '/user/output1';
OK
Time taken: 0.093 seconds
hive> INSERT OVERWRITE TABLE s1_p1 SELECT COUNT(*), v1.category
    >           FROM (
    >         SELECT
    >           CASE
    >           WHEN YEAR(t1.date) - YEAR(dob) <= 14 THEN 'children'
    >           WHEN YEAR(t1.date) - YEAR(dob) <= 24 THEN 'youth'
    >           WHEN YEAR(t1.date) - YEAR(dob) <= 64 THEN 'adults'
    >           ELSE 'senior citizen'
    >           END AS category,
    >           p.patientid AS patientid
    >           FROM Patient p
    >           INNER JOIN treatment t1 ON p.patientid = t1.patientid
    >           WHERE YEAR(t1.date) = 2022
    >           ) AS v1
    >           GROUP BY v1.category;
```

Creating a table in mysql:

```
create table first_problem_sol(count int, category varchar(50));
```

sqoop export:

```
sqoop export --connect jdbc:mysql://localhost:3306/output --username root --password cloudera -table first_problem_sol --export-dir /user/output1/000000_0 --input-fields-terminated-by ',';
```

```
^C[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/output --username root --password cloudera --table first_problem_sol --export-dir /user/output1/000000_0 --input-fields-terminated-by ',';
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
```

Output:

```
mysql> select * from first_problem_sol
    -> ;
+-------+----------------+
| count | category       |
+-------+----------------+
|    76 | youth          |
|  1404 | adults         |
|   788 | children       |
|   699 | senior citizen |
+-------+----------------+
4 rows in set (0.00 sec)
```

Problem statement 2: Jimmy, from the healthcare department, wants to know which disease is infecting people of which gender more often. Assist Jimmy with this purpose by generating a report that shows for each disease the male-to-female ratio. Sort the data in a way that is helpful for Jimmy.

```
hive> SELECT p.gender, v1.dname, COUNT(v1.dname) AS `Count`
    > FROM person p
    > JOIN
    > (
    >     SELECT t.patientid AS `pid`, t.diseaseid AS `did`, d.diseasename AS `dname`
    >     FROM treatment t
    >     JOIN disease d ON t.diseaseid = d.diseaseid
    > ) v1
    > ON p.personid = v1.pid
    > GROUP BY p.gender, v1.did, v1.dname;
```

Creating External Table:

```
    > create external table s2_p2 (diseasename varchar(50), malecount int, femalecount int, malefemale double)
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > location '/user/output2';
OK
Time taken: 0.338 seconds
hive> INSERT OVERWRITE TABLE s2_p2 select diseasename,COUNT(IF(gender = 'male', 1, null)) count_male,
    > COUNT(IF(gender = 'female', 1, NULL)) count_female,
    > COUNT(IF(gender = 'male', 1, NULL))/COUNT(IF(gender = 'female', 1, NULL)) as ratio
    > from
    > disease join treatment on treatment.diseaseid=disease.diseaseid
    > join patient on patient.patientid=treatment.patientid
    > join person on patient.patientid=person.personid
    > group by diseasename
    > order by diseasename ;
```

Creating a table in Mysql:

```
mysql> create database output;
Query OK, 1 row affected (0.00 sec)

mysql> use output;
Database changed
mysql>
mysql> create table second_problem_sol(diseasename varchar(50), malecount int, f
emalecount int, malefemale double);
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> show tables;
+--------------------+
| Tables_in_output   |
+--------------------+
| second_problem_sol |
+--------------------+
1 row in set (0.01 sec)
```

Sqoop export: sqoop export --connect jdbc:mysql://localhost:3306/output -username root --password cloudera --table second_problem_sol --export-dir /user/output2/000000_0 --input-fields-terminated-by ',' .

Output:

```
mysql> select * from second_problem_sol;
+----------------------------------------+-----------+-------------+----------
---------+
| diseasename                            | malecount | femalecount | malefemal
e       |
+----------------------------------------+-----------+-------------+----------
---------+
| Cancer                                 |       191 |         103 | 1.8543689
3203884 |
| Epilepsy                               |       153 |          96 |
1.59375 |
| Chronic fatigue syndrome               |       158 |         107 | 1.4766355
1401869 |
| Chronic obstructive pulmonary disease  |       152 |          97 | 1.5670103
0927835 |
| Coronary heart disease                 |       149 |          97 |  1.536082
4742268 |
| Crohn's disease                        |       182 |         102 |  1.784313
7254902 |
| Dementia                               |       162 |          90 |
    1.8 |
| Depression                             |       170 |          82 | 2.0731707
3170732 |
| Diabetes mellitus type 1               |       174 |          93 | 1.8709677
4193548 |
| Diabetes mellitus type 2               |       178 |          99 |  1.797979
7979798 |
| Dilated cardiomyopathy                 |       191 |         110 | 1.7363636
```

Problem Statement 3: Jacob, from insurance management, has noticed that insurance claims are not made for all the treatments. He also wants to figure out if the gender of the patient has any impact on the insurance claim. Assist Jacob in this situation by generating a report that finds for each gender the number of treatments, number of claims, and treatment-to-claim ratio. And notice if there is a significant difference between the treatment-to-claim ratio of male and female patients.

```
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive> SELECT v11.gender, v11.TCount, v22.CCount
    > FROM
    > (
    >     SELECT p.gender AS `gender`, COUNT(v1.did) AS `TCount`
    >     FROM person p
    >     JOIN
    >     (
    >         SELECT t.patientid AS `pid`, t.diseaseid AS `did`
    >         FROM treatment t
    >     ) AS v1
    >     ON p.personid = v1.pid
    >     GROUP BY p.gender
    > ) AS v11
    > JOIN
    > (
    >     SELECT p.gender AS `gender`, COUNT(v2.cid) AS `CCount`
    >     FROM person p
    >     JOIN
    >     (
    >         SELECT t.patientid AS `pid`, c.claimid AS `cid`
    >         FROM treatment t
    >         JOIN claim c ON t.claimid = c.claimid
    >     ) AS v2
    >     ON p.personid = v2.pid
    >     GROUP BY p.gender
    > ) AS v22
    > ON v11.gender = v22.gender;
```

Creating External Table:

```
hive>  create external table s3_p33(Gender varchar(20),Tcount int, Ccount int )
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > location '/user/output33';
OK
Time taken: 4.993 seconds
hive>
    > INSERT OVERWRITE TABLE s3_p33 SELECT v11.gender, v11.TCount, v22.CCount
    >       FROM
    >       (
    >           SELECT p.gender AS `gender`, COUNT(v1.did) AS `TCount`
    >           FROM person p
    >           JOIN
    >           (
    >               SELECT t.patientid AS `pid`, t.diseaseid AS `did`
    >               FROM treatment t
    >           ) AS v1
    >           ON p.personid = v1.pid
    >           GROUP BY p.gender
    >       ) AS v11
    >       JOIN
    >       (
    >           SELECT p.gender AS `gender`, COUNT(v2.cid) AS `CCount`
    >           FROM person p
    >           JOIN
    >           (
    >               SELECT t.patientid AS `pid`, c.claimid AS `cid`
    >               FROM treatment t
    >               JOIN claim c ON t.claimid = c.claimid
    >           ) AS v2
    >           ON p.personid = v2.pid
    >           GROUP BY p.gender
    >       ) AS v22
    >       ON v11.gender = v22.gender;
```

Creating a table in MySQL:

```
mysql> create table third_problem_solution(Gender varchar(20), Tcount int, Ccoun
t int);
Query OK, 0 rows affected (0.01 sec)
```

Sqoop export:

```
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/outp
ut --username root --password cloudera --table third_problem_solution --export-d
ir /user/output33/000000 0 --input-fields-terminated-by '.':
```

Output:

```
mysql> select * from third_problem_solution;
+--------+--------+--------+
| Gender | Tcount | Ccount |
+--------+--------+--------+
| female |   4206 |   2676 |
| male   |   6679 |   4287 |
+--------+--------+--------+
2 rows in set (0.00 sec)
```

Problem Statement 4: The Healthcare department wants a report about the inventory of pharmacies. Generate a report on their behalf that shows how many units of medicine each pharmacy has in their inventory, the total maximum retail price of those medicines, and the total price of all the medicines after discount. Note: discount field in keep signifies the percentage of discount on the maximum price.

```
hive> select a.pid as PharmacyID,sum(a.total),sum(a.after_discount) from (select k.pharmacyid as pid,(k.quantity*m.maxprice) as total,((k.quantity*m.maxprice
)-((k.quantity*m.maxprice)*k.discount/100)) as after_discount from pharmacy p join keep k on k.pharmacyid=p.pharmacyid join medicine m on m.medicineid=k.medi
cineid)a group by a.pid;
```

Creating a table in Mysql:

```
mysql> create table fourth_problem(pharmacy_id int, sum_count double, sum_discount double);
Query OK, 0 rows affected (0.02 sec)
```

Creating external table:

```
hive>
    > create external table s_p(pharmacy_id int, sum_count double, sum_discount double)
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > location '/user/output4a';
OK
Time taken: 4.008 seconds
hive>
    > insert overwrite s_p select a.pid as PharmacyID,sum(a.total),sum(a.after_discount) from (select k.pharmacyid as pid,(k.quantity*m.maxprice) as total,((
k.quantity*m.maxprice)-((k.quantity*m.maxprice)*k.discount/100)) as after_discount from pharmacy p join keep k on k.pharmacyid=p.pharmacyid join medicine m o
n m.medicineid=k.medicineid)a group by a.pid;
```

Sqoop export:

```
^C[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/output --username root --password cloudera --table fourth_problem --export-dir
ser/output4a/000000_0 --input-fields-terminated-by ','
```

Output:

```
mysql> select * from fourth_problem;
+-------------+------------------+------------------+
| pharmacy_id | sum_count        | sum_discount     |
+-------------+------------------+------------------+
|        7999 |     226608680.11 |    189533051.097 |
|        8109 |     750930577.12 | 614340282.164001 |
|        8142 |     550224876.46 |    419684104.771 |
|        8173 |    2308679004.64 |   1756426491.742 |
|        8184 |     978300503.93 |    795834270.359 |
|        8265 |     759569125.65 |    667423333.241 |
|        8315 |    1012065016.27 |    815419643.121 |
|        8320 |     988291697.76 |    845096027.213 |
|        8349 |       8832884.73 |      6622047.463 |
|        8404 |      50497526.59 |     39000143.377 |
|        8442 |     727710350.21 |    585930050.848 |
|        8549 |     635227157.13 |    544918987.613 |
|        8594 |     586623674.44 |    497282212.227 |
|        8628 |     1343228442.7 |   1005933000.441 |
|        8669 |     212973624.63 |    172986182.711 |
|        8718 |    2712101507.63 |   2100524119.165 |
|        8737 |     837092855.14 |    697260807.225 |
|        8760 |    1772426144.94 |   1535716954.383 |
|        8795 |      13851642.72 |      9799304.226 |
|        8824 |     859652251.98 |    698636798.485 |
|        8829 |     922977405.54 |    805653572.369 |
|        8852 |     188878443.71 |    179528621.888 |
|        8891 |     931223889.19 |    783035748.146 |
|        8897 |    1549309198.01 |   1348452929.882 |
|        8910 |     679575479.38 |    556269606.174 |
|        8911 |     950683603.29 |    856578605.991 |
|        8933 |    1055018554.23 |     906834476.94 |
|        8982 |     730842386.08 |    598901114.568 |
|        9010 |      427245328.9 |    360154035.372 |
|        9139 |       2392476.77 |       1853680.42 |
|        9169 |    1567601621.86 |   1292090260.401 |
|        9239 |     444952388.73 |    387084715.928 |
|        9255 |     244242979.94 |     211331961.22 |
```

Problem Statement 5: It is suspected by healthcare research department that the substance "ranitidine" might be causing some side effects. Find the top 3 companies using the substance in their medicine so that they can be informed about it.

select * from

(select dense_rank() over(partition by companyname order by quantity desc) as denseno,quantity,m.companyname as cname from keep join medicine m on keep.medicineid=m.medicineid where substancename='ranitidina')k where k.denseno=1 limit 3;

Creating external table:

```
hive> create external table s6_p1 (denseno int, quantity bigint, pharmacy varchar(20))
    > row format delimited
    > fields terminated by ','
    > lines terminated by '\n'
    > location '/user/output6';
OK
Time taken: 5.728 seconds
hive>  INSERT OVERWRITE TABLE s6_p1 select * from
    > (select dense_rank() over(partition by companyname order by quantity desc) as denseno,quantity,m.companyname as
 cname
    > from keep join medicine m on keep.medicineid=m.medicineid where substancename='ranitidina' )k
    > where k.denseno=1 limit 3;
Query ID = cloudera_20230315090000_319a7ea8-48de-4902-a75e-8f9ecea83fed
Total jobs = 2
```

Creating a table in mysql:

```
mysql> create table sixth_problem_sol(denseno int, quantity bigint, pharmacy varchar(20));
Query OK, 0 rows affected (0.02 sec)
```

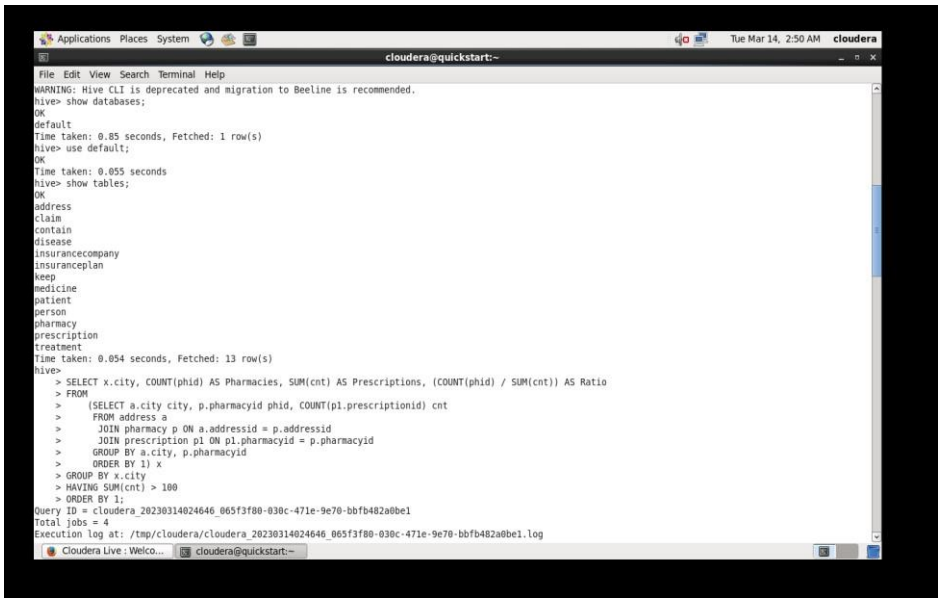
Sqoop Export:

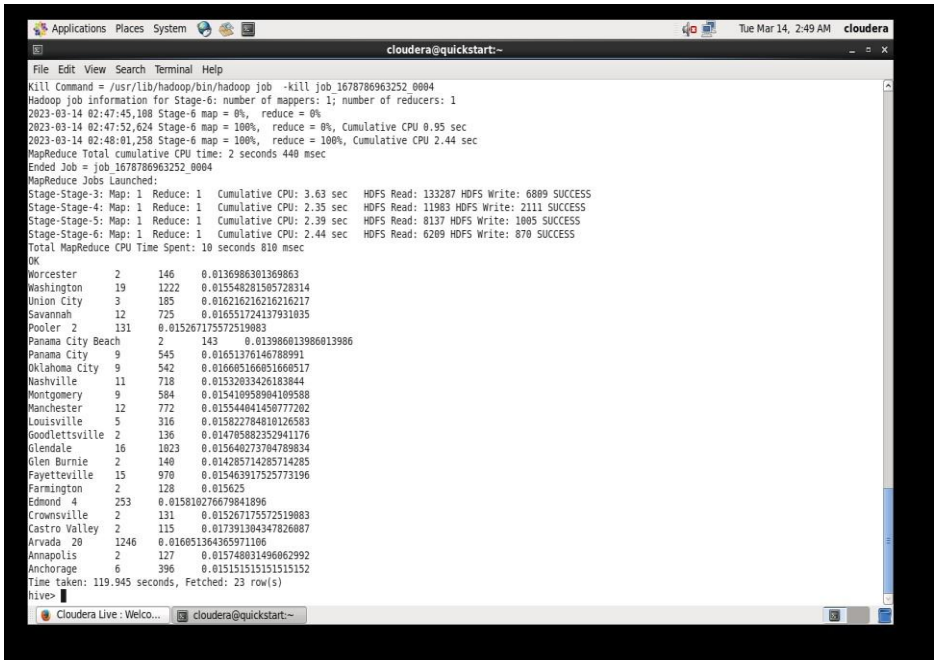
```
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost:3306/output --username root --password clouder
a --table sixth_problem_sol --export-dir /user/output6/000000_0 --input-fields-terminated-by ',';
```

Output:

```
mysql> select * from sixth_problem_sol;
+---------+----------+----------------------+
| denseno | quantity | pharmacy             |
+---------+----------+----------------------+
|       1 |     9686 | CIFARMA CIENTIFICA F |
|       1 |     6011 | BIOFARMA FARMACEUTIC |
|       1 |     9530 | HIPOLABOR FARMACEUTI |
+---------+----------+----------------------+
3 rows in set (0.00 sec)
```

Problem Statement 6: A company needs to set up 3 new pharmacies, they have come up with an idea that the pharmacy can be set up in cities where the pharmacy-to-prescription ratio is the lowest and the number of prescriptions should exceed 100. Assist the company to identify those cities where the pharmacy can be set up.





mysql> create table p_1(city varchar(20),Pharmacies int,Prescriptions int,Ratio double); CREATE EXTERNAL TABLE IF NOT EXISTS problem_1(city varchar(20),Pharmacies int,Prescriptions int,Ratio double)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

LOCATION '/user/output1';

```
File  Edit  View  Search  Terminal  Help
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.p
roperties
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive> show databases;
OK
default
Time taken: 1.374 seconds, Fetched: 1 row(s)
hive> use default;
OK
Time taken: 0.08 seconds
hive> CREATE EXTERNAL TABLE IF NOT EXISTS problem_1(city varchar(20),Pharmacies
int,Prescriptions int,Ratio double)
    > ROW FORMAT DELIMITED
    >  FIELDS TERMINATED BY ','
    >  LINES TERMINATED BY '\n'
    >  LOCATION '/user/output1';
OK
Time taken: 0.417 seconds
```

hive>

INSERT OVERWRITE TABLE problem_1 SELECT x.city, COUNT(phid) AS Pharmacies, SUM(cnt) AS Prescriptions, (COUNT(phid) / SUM(cnt)) AS Ratio

FROM

(SELECT a.city city, p.pharmacyid phid, COUNT(p1.prescriptionid) cnt

FROM address a

JOIN pharmacy p ON a.addressid = p.addressid

JOIN prescription p1 ON p1.pharmacyid = p.pharmacyid

GROUP BY a.city, p.pharmacyid

ORDER BY 1) x

GROUP BY x.city

HAVING SUM(cnt) > 100
ODER BY 1;

```
hive> INSERT OVERWRITE TABLE problem_1 SELECT x.city, COUNT(phid) AS Pharmacies,
 SUM(cnt) AS Prescriptions, (COUNT(phid) / SUM(cnt)) AS Ratio
    > FROM
    >     (SELECT a.city city, p.pharmacyid phid, COUNT(p1.prescriptionid) cnt
    >      FROM address a
    >       JOIN pharmacy p ON a.addressid = p.addressid
    >       JOIN prescription p1 ON p1.pharmacyid = p.pharmacyid
    >      GROUP BY a.city, p.pharmacyid
    >      ORDER BY 1) x
    > GROUP BY x.city
    > HAVING SUM(cnt) > 100
    > ORDER BY 1;
Query ID = cloudera_20230315091010_8cd99ff4-d513-4cab-9980-9bbefa125a69
Total jobs = 4
Execution log at: /tmp/cloudera/cloudera_20230315091010_8cd99ff4-d513-4cab-9980-
9bbefa125a69.log
2023-03-15 09:10:25     Starting to launch local task to process map join;     m
aximum memory = 1013645312
2023-03-15 09:10:29     Dump the side-table for tag: 1 with group count: 213 int
o file: file:/tmp/cloudera/7146e199-c1ca-4e72-af6f-80627c31dc43/hive_2023-03-15_
09-10-08_017_4273390198409051605-1/-local-10007/HashTable-Stage-3/MapJoin-mapfil
e01--.hashtable
2023-03-15 09:10:30     Uploaded 1 File to: file:/tmp/cloudera/7146e199-c1ca-4e7
2-af6f-80627c31dc43/hive_2023-03-15_09-10-08_017_4273390198409051605-1/-local-10
007/HashTable-Stage-3/MapJoin-mapfile01--.hashtable (147237 bytes)
2023-03-15 09:10:30     Dump the side-table for tag: 1 with group count: 213 int
o file: file:/tmp/cloudera/7146e199-c1ca-4e72-af6f-80627c31dc43/hive_2023-03-15_
09-10-08_017_4273390198409051605-1/-local-10007/HashTable-Stage-3/MapJoin-mapfil
e11--.hashtable
2023-03-15 09:10:30     Uploaded 1 File to: file:/tmp/cloudera/7146e199-c1ca-4e7
2-af6f-80627c31dc43/hive_2023-03-15_09-10-08_017_4273390198409051605-1/-local-10
007/HashTable-Stage-3/MapJoin-mapfile11--.hashtable (5610 bytes)
2023-03-15 09:10:30     End of local task; Time Taken: 4.599 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 4
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
```

sqoop export --connect jdbc:mysql://localhost:3306/solution --username root --password cloudera -table p_1 --export-dir /user/output1/000000_0 --input-fields-terminated-by ','; **Output:**

```
mysql> select * from p_1;
+-------------------+------------+---------------+--------------------+
| city              | Pharmacies | Prescriptions | Ratio              |
+-------------------+------------+---------------+--------------------+
| Crownsville       |          2 |           131 | 0.0152671755725191 |
| Worcester         |          2 |           146 | 0.0136986301369863 |
| Washington        |         19 |          1222 | 0.0155482815057283 |
| Union City        |          3 |           185 | 0.0162162162162162 |
| Savannah          |         12 |           725 |  0.016551724137931 |
| Pooler            |          2 |           131 | 0.0152671755725191 |
| Panama City Beach |          2 |           143 |  0.013986013986014 |
| Castro Valley     |          2 |           115 | 0.0173913043478261 |
| Arvada            |         20 |          1246 | 0.0160513643659711 |
| Annapolis         |          2 |           127 |  0.015748031496063 |
| Anchorage         |          6 |           396 | 0.0151515151515152 |
| Panama City       |          9 |           545 | 0.0165137614678899 |
| Oklahoma City     |          9 |           542 | 0.0166051660516605 |
| Nashville         |         11 |           718 | 0.0153203342618384 |
| Montgomery        |          9 |           584 | 0.0154109589041096 |
| Manchester        |         12 |           772 | 0.0155440414507772 |
| Louisville        |          5 |           316 | 0.0158227848101266 |
| Goodlettsville    |          2 |           136 | 0.0147058823529412 |
| Glendale          |         16 |          1023 | 0.0156402737047898 |
| Glen Burnie       |          2 |           140 | 0.0142857142857143 |
| Fayetteville      |         15 |           970 | 0.0154639175257732 |
| Farmington        |          2 |           128 |              0.015625 |
| Edmond            |          4 |           253 | 0.0158102766798419 |
+-------------------+------------+---------------+--------------------+
23 rows in set (0.00 sec)

mysql>
```

Problem Statement 7: The State of Alabama (AL) is trying to manage its healthcare resources more efficiently. For each city in their state, they need to identify the disease for which the maximum number of patients have gone for treatment. Assist the state for this purpose. Note: The state of Alabama is represented as AL in Address Table.

```
hive> SELECT city, diseaseName, treat_count
    > FROM (
    >     SELECT *,
    >         DENSE_RANK() OVER (PARTITION BY city ORDER BY treat_count DESC) AS rank1
    >     FROM (
    >         SELECT city, diseaseName, COUNT(treatmentID) AS treat_count
    >         FROM address
    >         INNER JOIN person ON address.addressID = person.addressID
    >         INNER JOIN patient ON person.personID = patient.patientID
    >         INNER JOIN treatment ON patient.patientID = treatment.patientID
    >         INNER JOIN disease ON treatment.diseaseID = disease.diseaseID
    >         WHERE state = 'AL'
    >         GROUP BY city, diseaseName
    >     ) AS tb
    > ) AS tre
    > WHERE rank1 = 1
    > ORDER BY treat_count DESC;
Query ID = cloudera_20230314040505_97a200d2-6e7d-4d37-98c9-abfae73d09a3
Total jobs = 3
Execution log at: /tmp/cloudera/cloudera_20230314040505_97a200d2-6e7d-4d37-98c9-abfae73d09a3.log
2023-03-14 04:05:54    Starting to launch local task to process map join;      maximum memory = 1013645312
2023-03-14 04:05:56    Dump the side-table for tag: 1 with group count: 40 into file: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-mapf
ile01--.hashtable
2023-03-14 04:05:56    Uploaded 1 File to: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-mapfile01--.hashtable (1754 bytes)
2023-03-14 04:05:56    Dump the side-table for tag: 1 with group count: 1126 into file: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-ma
pfile11--.hashtable
2023-03-14 04:05:56    Uploaded 1 File to: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-mapfile11--.hashtable (24021 bytes)
2023-03-14 04:05:56    Dump the side-table for tag: 2 with group count: 1052 into file: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-ma
pfile12--.hashtable
2023-03-14 04:05:56    Uploaded 1 File to: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-mapfile12--.hashtable (127612 bytes)
2023-03-14 04:05:56    Dump the side-table for tag: 1 with group count: 1673 into file: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-ma
pfile21--.hashtable
2023-03-14 04:05:56    Uploaded 1 File to: file:/tmp/cloudera/fc0f2d84-cef4-4955-aa9c-a620950b455f/hive_2023-03-14_04-05-46_452_2834271308427385047-1/-local-10009/HashTable-Stage-4/MapJoin-mapfile21--.hashtable (53061 bytes)
2023-03-14 04:05:56    End of local task; Time Taken: 2.898 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 3


In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1678786963252_0016, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1678786963252_0016/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1678786963252_0016
Hadoop job information for Stage-6: number of mappers: 1; number of reducers: 1
2023-03-14 04:06:57,091 Stage-6 map = 0%,  reduce = 0%
2023-03-14 04:07:03,852 Stage-6 map = 100%,  reduce = 0%, Cumulative CPU 0.71 sec
2023-03-14 04:07:11,492 Stage-6 map = 100%,  reduce = 100%, Cumulative CPU 1.98 sec
MapReduce Total cumulative CPU time: 1 seconds 980 msec
Ended Job = job_1678786963252_0016
MapReduce Jobs Launched:
Stage-Stage-4: Map: 1  Reduce: 1   Cumulative CPU: 3.14 sec   HDFS Read: 138844 HDFS Write: 2700 SUCCESS
Stage-Stage-5: Map: 1  Reduce: 1   Cumulative CPU: 2.34 sec   HDFS Read: 9643 HDFS Write: 590 SUCCESS
Stage-Stage-6: Map: 1  Reduce: 1   Cumulative CPU: 1.98 sec   HDFS Read: 5318 HDFS Write: 354 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 460 msec
OK
Montgomery      Cancer  28
Montgomery      Guillain?Barré syndrome 28
Montevallo      Schizophrenia   2
Indian Springs Village  Bipolar disorder        1
Indian Springs Village  Schizophrenia   1
Indian Springs Village  Parkinson's disease     1
Indian Springs Village  Multiple sclerosis      1
Indian Springs Village  Alzheimer's disease     1
Indian Springs Village  Diabetes mellitus type 2        1
Time taken: 87.203 seconds, Fetched: 9 row(s)
hive>
```

create table p_2(city varchar(25),diseaseName varchar(25),treat_count int);

```
CREATE EXTERNAL TABLE IF NOT EXISTS problem_2 (city varchar(25),diseaseName
varchar(25),treat_count int)
 ROW FORMAT DELIMITED
 FIELDS TERMINATED BY ','
 LINES TERMINATED BY '\n'
 LOCATION '/user/output2';
```

```
INSERT OVERWRITE TABLE problem_2 SELECT city, diseaseName, treat_count
    FROM (
      SELECT *,
        DENSE_RANK() OVER (PARTITION BY city ORDER BY treat_count DESC) AS rank1
      FROM (
        SELECT city, diseaseName, COUNT(treatmentID) AS treat_count
FROM address
        INNER JOIN person ON address.addressID = person.addressID
        INNER JOIN patient ON person.personID = patient.patientID
        INNER JOIN treatment ON patient.patientID = treatment.patientID
        INNER JOIN disease ON treatment.diseaseID = disease.diseaseID
        WHERE state = 'AL'
        GROUP BY city, diseaseName
      ) AS tb
    ) AS tre
    WHERE rank1 = 1
    ORDER BY treat_count DESC;
```

```
Time taken: 190.162 seconds, Fetched: 9 row(s)
hive> CREATE EXTERNAL TABLE IF NOT EXISTS problem_2 (city varchar(25),diseaseName
    > varchar(25),treat_count int)
    >  ROW FORMAT DELIMITED
    >  FIELDS TERMINATED BY ','
    >  LINES TERMINATED BY '\n'
    >  LOCATION '/user/output2';
OK
Time taken: 0.24 seconds
hive> INSERT OVERWRITE TABLE problem_2 SELECT city, diseaseName, treat_count
    >       FROM (
    >           SELECT *,
    >               DENSE_RANK() OVER (PARTITION BY city ORDER BY treat_count DESC) AS rank1
    >           FROM (
    >               SELECT city, diseaseName, COUNT(treatmentID) AS treat_count
    >               FROM address
    >               INNER JOIN person ON address.addressID = person.addressID
    >               INNER JOIN patient ON person.personID = patient.patientID
    >               INNER JOIN treatment ON patient.patientID = treatment.patientID
    >               INNER JOIN disease ON treatment.diseaseID = disease.diseaseID
    >               WHERE state = 'AL'
    >               GROUP BY city, diseaseName
    >           ) AS tb
    >       ) AS tre
    >       WHERE rank1 = 1
    >       ORDER BY treat_count DESC;
Query ID = cloudera_20230315084646_834bb481-a192-49f5-a0c0-d0c7867daf76
Total jobs = 3
Execution log at: /tmp/cloudera/cloudera_20230315084646_834bb481-a192-49f5-a0c0-d0c7867daf76.log
2023-03-15 08:46:50     Starting to launch local task to process map join;      maximum memory =
```

sqoop export --connect jdbc:mysql://localhost:3306/solution --username root --password cloudera -table p_2 --export-dir /user/output2/000000_0 --input-fields-terminated-by ',';

output:

```
Database changed
mysql> show tables;
+---------------------+
| Tables_in_solution |
+---------------------+
| p_2                 |
| p_5                 |
+---------------------+
2 rows in set (0.00 sec)

mysql> select * from p_2;
+--------------------------+-----------------------------+--------------+
| city                     | diseaseName                 | treat_count |
+--------------------------+-----------------------------+--------------+
| Indian Springs Village   | Multiple sclerosis          |            1 |
| Indian Springs Village   | Schizophrenia               |            1 |
| Indian Springs Village   | Alzheimer's disease         |            1 |
| Montgomery               | Cancer                      |           28 |
| Montgomery               | Guillain?Barré syndrome     |           28 |
| Montevallo               | Schizophrenia               |            2 |
| Indian Springs Village   | Bipolar disorder            |            1 |
| Indian Springs Village   | Parkinson's disease         |            1 |
| Indian Springs Village   | Diabetes mellitus type 2    |            1 |
+--------------------------+-----------------------------+--------------+
9 rows in set (0.00 sec)

mysql>
```

Problem Statement 8: An Insurance company wants a state wise report of the treatments to claim ratio between 1st April 2021 and 31st March 2022 (days both included).

Assist them to create such a report.

```
Time taken: 31.992 seconds, Fetched: 1781 row(s)
hive> SELECT address.state, COUNT(treatment.treatmentID) AS treat_count,
    >     COUNT(claim.claimID) AS claim_count,
    >     COUNT(treatment.treatmentID) / COUNT(claim.claimID) AS ratio
    > FROM address
    > INNER JOIN person ON address.addressID = person.addressID
    > INNER JOIN patient ON person.personID = patient.patientID
    > INNER JOIN treatment ON patient.patientID = treatment.patientID
    > LEFT JOIN claim ON treatment.claimID = claim.claimID
    > WHERE treatment.date BETWEEN '2021-04-01' AND '2022-03-31'
    > GROUP BY address.state;
Query ID = cloudera_20230314042323_2d99be60-74e2-4973-bd0d-884ed96f4eab
Total jobs = 1
Execution log at: /tmp/cloudera/cloudera_20230314042323_2d99be60-74e2-4973-bd0d-884ed96f4eab.log
2023-03-14 04:23:34     Starting to launch local task to process map join;      maximum memory = 1013645312
2023-03-14 04:23:36     Dump the side-table for tag: 1 with group count: 6963 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/
pfile41--.hashtable
2023-03-14 04:23:36     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfi
2023-03-14 04:23:36     Dump the side-table for tag: 1 with group count: 1126 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/
pfile51--.hashtable
2023-03-14 04:23:36     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfi
2023-03-14 04:23:36     Dump the side-table for tag: 2 with group count: 819 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/
file52--.hashtable
2023-03-14 04:23:36     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfi
2023-03-14 04:23:36     Dump the side-table for tag: 1 with group count: 1673 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/
pfile61--.hashtable
2023-03-14 04:23:37     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfi
2023-03-14 04:23:37     End of local task; Time Taken: 2.277 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
```

```
2023-03-14 04:23:36     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile51--.hashtable (24021 bytes)
2023-03-14 04:23:36     Dump the side-table for tag: 2 with group count: 819 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-map
file52--.hashtable
2023-03-14 04:23:36     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile52--.hashtable (49400 bytes)
2023-03-14 04:23:36     Dump the side-table for tag: 1 with group count: 1673 into file: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-ma
pfile61--.hashtable
2023-03-14 04:23:37     Uploaded 1 File to: file:/tmp/cloudera/401183fc-c90a-4607-ba06-783469f4b7fd/hive_2023-03-14_04-23-29_284_9099062249287939852-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile61--.hashtable (53061 bytes)
2023-03-14 04:23:37     End of local task; Time Taken: 2.277 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1678786963252_0018, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1678786963252_0018/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1678786963252_0018
Hadoop job information for Stage-4: number of mappers: 1; number of reducers: 1
2023-03-14 04:23:46,252 Stage-4 map = 0%,  reduce = 0%
2023-03-14 04:23:54,948 Stage-4 map = 100%,  reduce = 0%, Cumulative CPU 2.83 sec
2023-03-14 04:24:03,602 Stage-4 map = 100%,  reduce = 100%, Cumulative CPU 3.74 sec
MapReduce Total cumulative CPU time: 3 seconds 740 msec
Ended Job = job_1678786963252_0018
MapReduce Jobs Launched:
Stage-Stage-4: Map: 1  Reduce: 1   Cumulative CPU: 3.74 sec   HDFS Read: 138448 HDFS Write: 466 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 740 msec
OK
AK      98      67      1.462686567164179
AL      213     130     1.6384615384615384
AR      141     92      1.5326086956521738
AZ      135     82      1.6463414634146434
CA      267     182     1.4678329670329967
CO      182     114     1.5964912280701755
CT      196     135     1.4518518518518052
DC      167     110     1.5181818181818018
FL      192     114     1.6842105263157894
GA      195     127     1.5354330708661417
KY      128     67      1.4712643678160092
MA      142     96      1.4791666666666667
MD      167     110     1.5181818181818018
OK      207     123     1.6829268292682926
TN      208     123     1.6910569105691058
VT      131     89      1.4719101123595506
Time taken: 35.42 seconds, Fetched: 16 row(s)
hive>
    >
```

create table p_5(state varchar(25),treatcount int,claimcount int,ratio double);

CREATE EXTERNAL TABLE IF NOT EXISTS problem_5 (state varchar(25),treatcount int,claimcount int,ratio double)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
LOCATION '/user/output';

INSERT OVERWRITE TABLE problem_5 SELECT address.state, COUNT(treatment.treatmentID) AS treat_count,
COUNT(claim.claimID) AS claim_count,
COUNT(treatment.treatmentID) / COUNT(claim.claimID) AS ratio
FROM address
INNER JOIN person ON address.addressID = person.addressID
INNER JOIN patient ON person.personID = patient.patientID
INNER JOIN treatment ON patient.patientID = treatment.patientID
LEFT JOIN claim ON treatment.claimID = claim.claimID
WHERE treatment.date BETWEEN '2021-04-01' AND '2022-03-31'
GROUP BY address.state;

```
cloudera@quickstart:~
File  Edit  View  Search  Terminal  Help
hive> CREATE EXTERNAL TABLE IF NOT EXISTS problem_5 (state varchar(25),treatcount int,claimcount int,ratio double)
    >  ROW FORMAT DELIMITED
    >  FIELDS TERMINATED BY ','
    >  LINES TERMINATED BY '\n'
    >  LOCATION '/user/output';
OK
Time taken: 0.317 seconds
hive> INSERT OVERWRITE TABLE problem_5 SELECT address.state, COUNT(treatment.treatmentID) AS treat_count,
    >     COUNT(claim.claimID) AS claim_count,
    >     COUNT(treatment.treatmentID) / COUNT(claim.claimID) AS ratio
    > FROM address
    > INNER JOIN person ON address.addressID = person.addressID
    > INNER JOIN patient ON person.personID = patient.patientID
    > INNER JOIN treatment ON patient.patientID = treatment.patientID
    > LEFT JOIN claim ON treatment.claimID = claim.claimID
    > WHERE treatment.date BETWEEN '2021-04-01' AND '2022-03-31'
    > GROUP BY address.state;
Query ID = cloudera_20230315052222_a4c44f20-e90c-4c30-b446-7282fe28fb6d
Total jobs = 1
Execution log at: /tmp/cloudera/cloudera_20230315052222_a4c44f20-e90c-4c30-b446-7282fe28fb6d.log
2023-03-15 05:22:52     Starting to launch local task to process map join;      maximum memory = 1013645312
2023-03-15 05:22:57     Dump the side-table for tag: 1 with group count: 6963 into file: file:/tmp/cloudera/5c0296a
2692-4e29-8761-f40e8302d35c/hive_2023-03-15_05-22-34_310_6733625594716823598-1/-local-10005/HashTable-Stage-4/MapJo
-mapfile01--.hashtable
2023-03-15 05:22:57     Uploaded 1 File to: file:/tmp/cloudera/5c0296a6-2692-4e29-8761-f40e8302d35c/hive_2023-03-15
5-22-34_310_6733625594716823598-1/-local-10005/HashTable-Stage-4/MapJoin-mapfile01--.hashtable (158665 bytes)
2023-03-15 05:22:57     Dump the side-table for tag: 1 with group count: 1126 into file: file:/tmp/cloudera/5c0296a
2692-4e29-8761-f40e8302d35c/hive_2023-03-15_05-22-34_310_6733625594716823598-1/-local-10005/HashTable-Stage-4/MapJo
```

sqoop export --connect jdbc:mysql://localhost:3306/solution --username root --password cloudera -table p_5 --export-dir /user/output/000000_0 --input-fields-terminated-by ',';

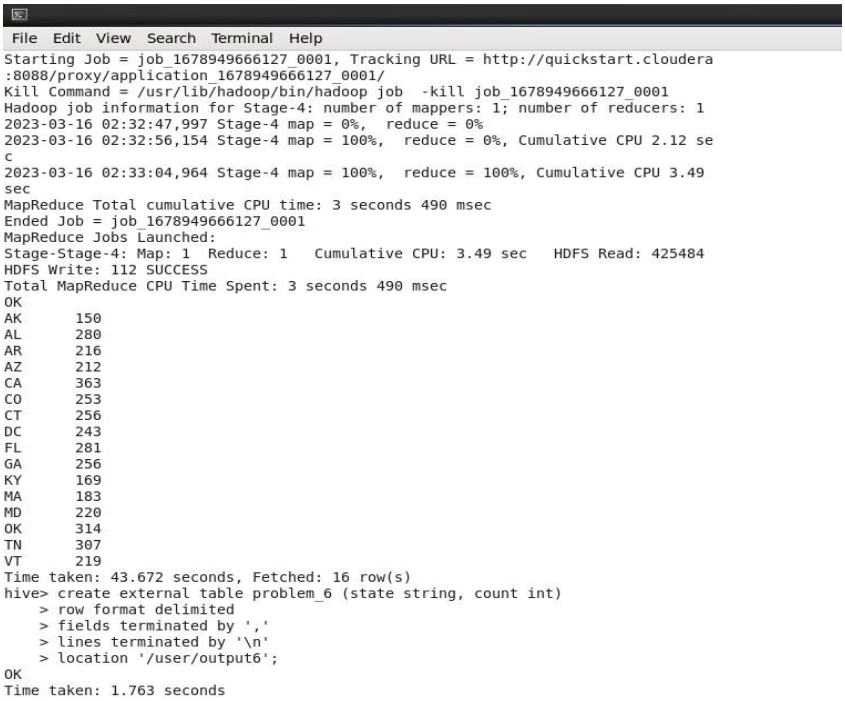
Output:

```
mysql> use solution;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+--------------------+
| Tables_in_solution |
+--------------------+
| p2_5               |
+--------------------+
1 row in set (0.00 sec)

mysql> select * from p2_5;
ERROR 1146 (42S02): Table 'solution.p2_5' doesn't exist
mysql> show tables;
+--------------------+
| Tables_in_solution |
+--------------------+
| p_5                |
+--------------------+
1 row in set (0.00 sec)

mysql> select * from p_5;
+-------+------------+------------+------------------+
| state | treatcount | claimcount | ratio            |
+-------+------------+------------+------------------+
| AK    |         98 |         67 | 1.46268656716418 |
| OK    |        207 |        123 | 1.68292682926829 |
| TN    |        208 |        123 | 1.69105691056911 |
| VT    |        131 |         89 | 1.47191011235955 |
| AL    |        213 |        130 | 1.63846153846154 |
| AR    |        141 |         92 | 1.53260869565217 |
| AZ    |        135 |         82 | 1.64634146341463 |
| CA    |        267 |        182 | 1.46703296703297 |
| CO    |        182 |        114 | 1.59649122807018 |
| CT    |        196 |        135 | 1.45185185185185 |
| DC    |        167 |        110 | 1.51818181818182 |
| FL    |        192 |        114 | 1.68421052631579 |
| GA    |        195 |        127 | 1.53543307086614 |
| KY    |        128 |         87 | 1.47126436781609 |
| MA    |        142 |         96 | 1.47916666666667 |
| MD    |        167 |        110 | 1.51818181818182 |
+-------+------------+------------+------------------+
16 rows in set (0.00 sec)
```

Problem statement 9: Manish, from the healthcare department, wants to know how many registered people are registered as patients as well, in each city. Generate a report that shows each city that has 10 or more registered people belonging to it and the number of patients from that city as well as the percentage of the patient with respect to the registered people.
create table p_6(state string, count int);

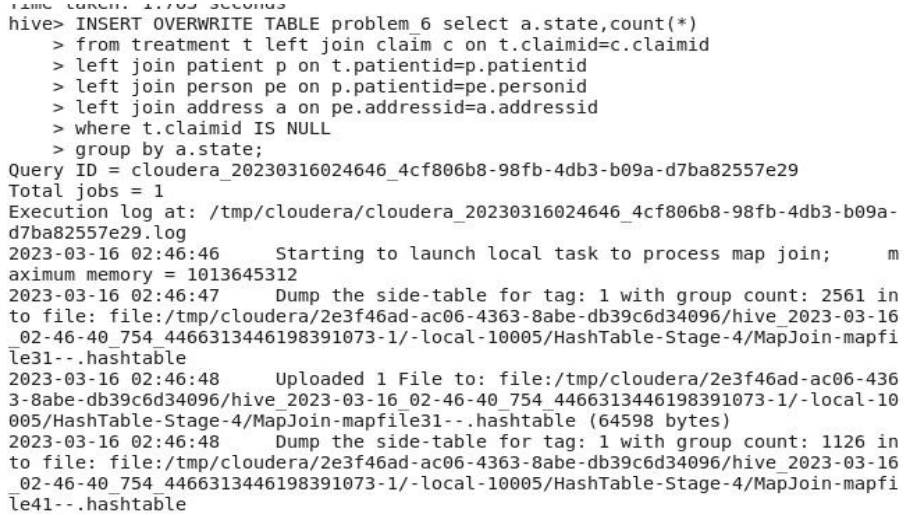
```
fix-tree module jar containing PrefixTreeCodec is not present.  Continuing witho
ut it.

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.p
roperties
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive>
    > ;
hive> select a.state,count(*)
    > from treatment t left join claim c on t.claimid=c.claimid
    > left join patient p on t.patientid=p.patientid
    > left join person pe on p.patientid=pe.personid
    > left join address a on pe.addressid=a.addressid
    > where t.claimid IS NULL
    > group by a.state;
Query ID = cloudera_20230316023232_af13d2d2-ed33-494f-a440-84dc1edb5932
Total jobs = 1
Execution log at: /tmp/cloudera/cloudera_20230316023232_af13d2d2-ed33-494f-a440-
84dc1edb5932.log
```

create external table problem_6(state string, count int) row format delimited fields terminated by ',' lines terminated by '\n' location '/user/output/output6';



INSERT OVERWRITE TABLE problem_6 select a.state,count(*) from treatment t left join claim c on t.claimid=c.claimid left join patient p on t.patientid=p.patientid left join person pe on p.patientid=pe.personid left join address a on pe.addressid=a.addressid where t.claimid IS NULL group by a.state;

sqoop export --connect jdbc:mysql://localhost:3306/solution --username root --password cloudera -table p_6 --export-dir /user/output6/s6_p4/000000_0 --input-fields-terminated-by ',';



Output:

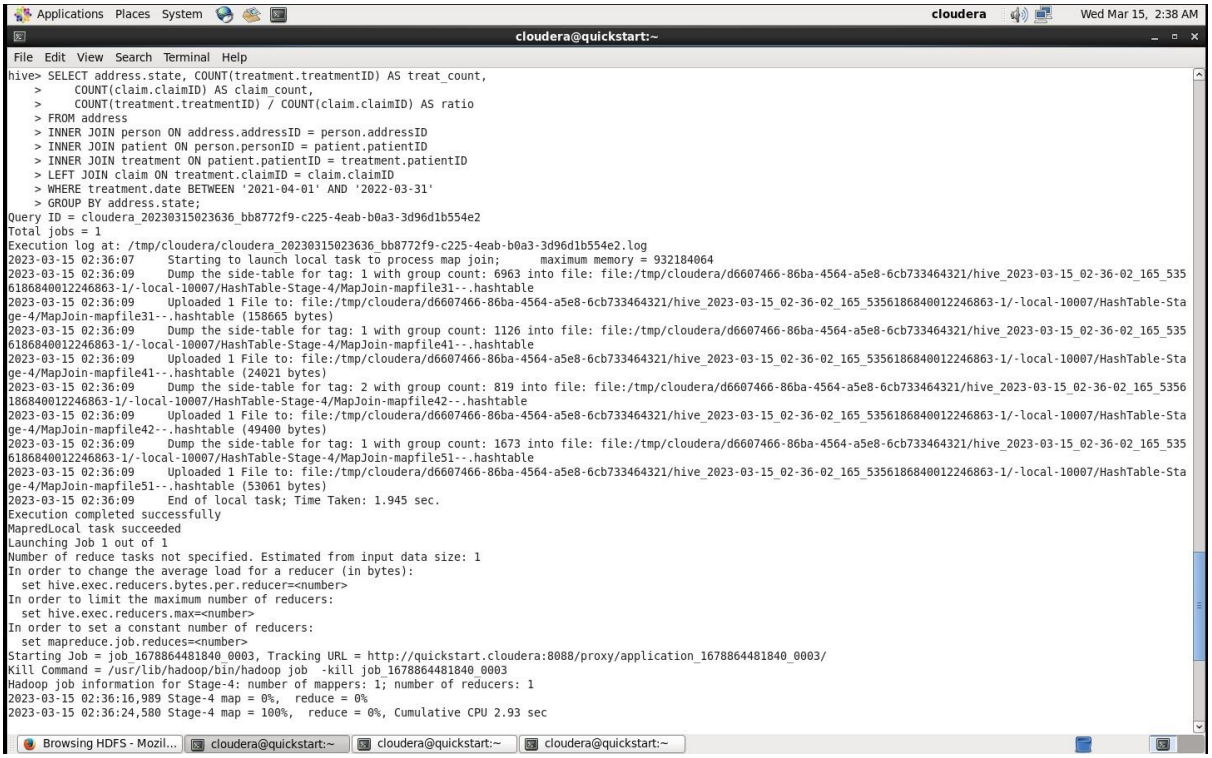
```
mysql> create table p_6(state varchar(20),count int);
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
+--------------------+
| Tables_in_solution |
+--------------------+
| p_1                |
| p_2                |
| p_5                |
| p_6                |
+--------------------+
4 rows in set (0.01 sec)

mysql> select * from p_6;
+-------+-------+
| state | count |
+-------+-------+
| AK    |   150 |
| AL    |   280 |
| AR    |   216 |
| AZ    |   212 |
| CA    |   363 |
| CO    |   253 |
| CT    |   256 |
| DC    |   243 |
| FL    |   281 |
| GA    |   256 |
| KY    |   169 |
| MA    |   183 |
| MD    |   220 |
| OK    |   314 |
| TN    |   307 |
| VT    |   219 |
+-------+-------+
16 rows in set (0.00 sec)

mysql>
```

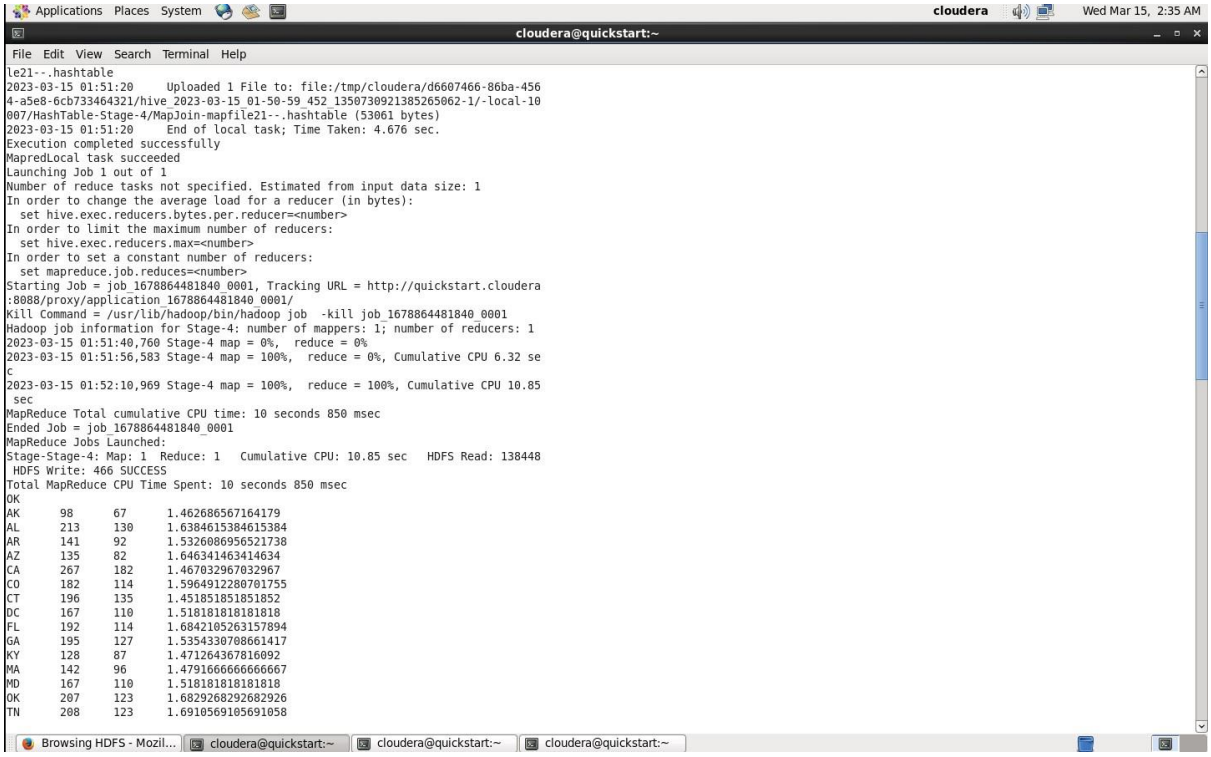
Problem statement 10: An Insurance company wants a state wise report of the treatments to claim ratio between 1st April 2021 and 31st March 2022 (days both included). Assist them to create such a report.

SELECT address.state, COUNT(treatment.treatmentID) AS treat_count, COUNT(claim.claimID) AS claim_count, COUNT(treatment.treatmentID) / COUNT(claim.claimID) AS ratioFROM addressINNER JOIN person ON address.addressID = person.addressIDINNER JOIN patient ON person.personID = patient.patientIDINNER JOIN treatment ON patient.patientID = treatment.patientIDLEFT JOIN claim ON treatment.claimID = claim.claimIDWHERE treatment.date BETWEEN '2021-04-01' AND '2022-0331'GROUP BY address.state;

create table address_part1 (addressid int , address1 string, city string, zip int) partitioned by (state string);

insert into address_part1 partition(state) select addressid, address1,city, zip,state from address;


Applications Places System
cloudera
Wed Mar 15, 2:38 AM
cloudera@quickstart:~
File Edit View Search Terminal Help
hive> SELECT address.state, COUNT(treatment.treatmentID) AS treat_count,
> COUNT(claim.claimID) AS claim_count,
> COUNT(treatment.treatmentID) / COUNT(claim.claimID) AS ratio
> FROM address
> INNER JOIN person ON address.addressID = person.addressID
> INNER JOIN patient ON person.personID = patient.patientID
> INNER JOIN treatment ON patient.patientID = treatment.patientID
> LEFT JOIN claim ON treatment.claimID = claim.claimID
> WHERE treatment.date BETWEEN '2021-04-01' AND '2022-03-31'
> GROUP BY address.state;
Query ID = cloudera_20230315023636_bb8772f9-c225-4eab-b0a3-3d96d1b554e2
Total jobs = 1
Execution log at: /tmp/cloudera/cloudera_20230315023636_bb8772f9-c225-4eab-b0a3-3d96d1b554e2.log
2023-03-15 02:36:07 Starting to launch local task to process map join; maximum memory = 932184064
2023-03-15 02:36:09 Dump the side-table for tag: 1 with group count: 6963 into file: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile31--.hashtable
2023-03-15 02:36:09 Uploaded 1 File to: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile31--.hashtable (158665 bytes)
2023-03-15 02:36:09 Dump the side-table for tag: 1 with group count: 1126 into file: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile41--.hashtable
2023-03-15 02:36:09 Uploaded 1 File to: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile41--.hashtable (24021 bytes)
2023-03-15 02:36:09 Dump the side-table for tag: 2 with group count: 819 into file: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile42--.hashtable
2023-03-15 02:36:09 Uploaded 1 File to: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile42--.hashtable (49400 bytes)
2023-03-15 02:36:09 Dump the side-table for tag: 1 with group count: 1673 into file: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile51--.hashtable
2023-03-15 02:36:09 Uploaded 1 File to: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_02-36-02_165_5356186840012246863-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile51--.hashtable (53061 bytes)
2023-03-15 02:36:09 End of local task; Time Taken: 1.945 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
Starting Job = job_1678864481840_0003, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1678864481840_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1678864481840_0003
Hadoop job information for Stage-4: number of mappers: 1; number of reducers: 1
2023-03-15 02:36:16,989 Stage-4 map = 0%, reduce = 0%
2023-03-15 02:36:24,580 Stage-4 map = 100%, reduce = 0%, Cumulative CPU 2.93 sec
Browsing HDFS - Mozil...
cloudera@quickstart:~
cloudera@quickstart:~
cloudera@quickstart:~


Applications Places System
cloudera
Wed Mar 15, 2:35 AM
cloudera@quickstart:~
File Edit View Search Terminal Help
le21--.hashtable
2023-03-15 01:51:20 Uploaded 1 File to: file:/tmp/cloudera/d6607466-86ba-4564-a5e8-6cb733464321/hive_2023-03-15_01-50-59_452_1350730921385265062-1/-local-10007/HashTable-Stage-4/MapJoin-mapfile21--.hashtable (53061 bytes)
2023-03-15 01:51:20 End of local task; Time Taken: 4.676 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
Starting Job = job_1678864481840_0001, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1678864481840_0001/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1678864481840_0001
Hadoop job information for Stage-4: number of mappers: 1; number of reducers: 1
2023-03-15 01:51:40,760 Stage-4 map = 0%, reduce = 0%
2023-03-15 01:51:56,583 Stage-4 map = 100%, reduce = 0%, Cumulative CPU 6.32 sec
2023-03-15 01:52:10,969 Stage-4 map = 100%, reduce = 100%, Cumulative CPU 10.85 sec
MapReduce Total cumulative CPU time: 10 seconds 850 msec
Ended Job = job_1678864481840_0001
MapReduce Jobs Launched:
Stage-Stage-4: Map: 1 Reduce: 1 Cumulative CPU: 10.85 sec HDFS Read: 138448 HDFS Write: 466 SUCCESS
Total MapReduce CPU Time Spent: 10 seconds 850 msec
OK
AK 98 67 1.462686567164179
AL 213 130 1.6384615384615384
AR 141 92 1.5326086956521738
AZ 135 82 1.646341463414634
CA 267 182 1.467032967032967
CO 182 114 1.5964912280701755
CT 196 135 1.451851851851852
DC 167 110 1.518181818181818
FL 192 114 1.6842105263157894
GA 195 127 1.5354330708661417
KY 128 87 1.471264367816092
MA 142 96 1.4791666666666667
MD 167 110 1.518181818181818
OK 207 123 1.6829268292682926
TN 208 123 1.6910569105691058
Browsing HDFS - Mozil...
cloudera@quickstart:~
cloudera@quickstart:~
cloudera@quickstart:~