CSEE5590 Big Data Programming

In Class Programming –5 Report
(Jongkook Son)

Project Overview:

Sqoop is a transfer tool between Hadoop and SQL or Relational Databases

Requirements/Task(s):

1. Use Sqoop to import and export mySQL Tables to HDFS.

2. Create Hive Tables through HQL Script, Use Sqoop to import and export tables
   to Relational Databases

3. Perform three queries from databases

See the slides for more details

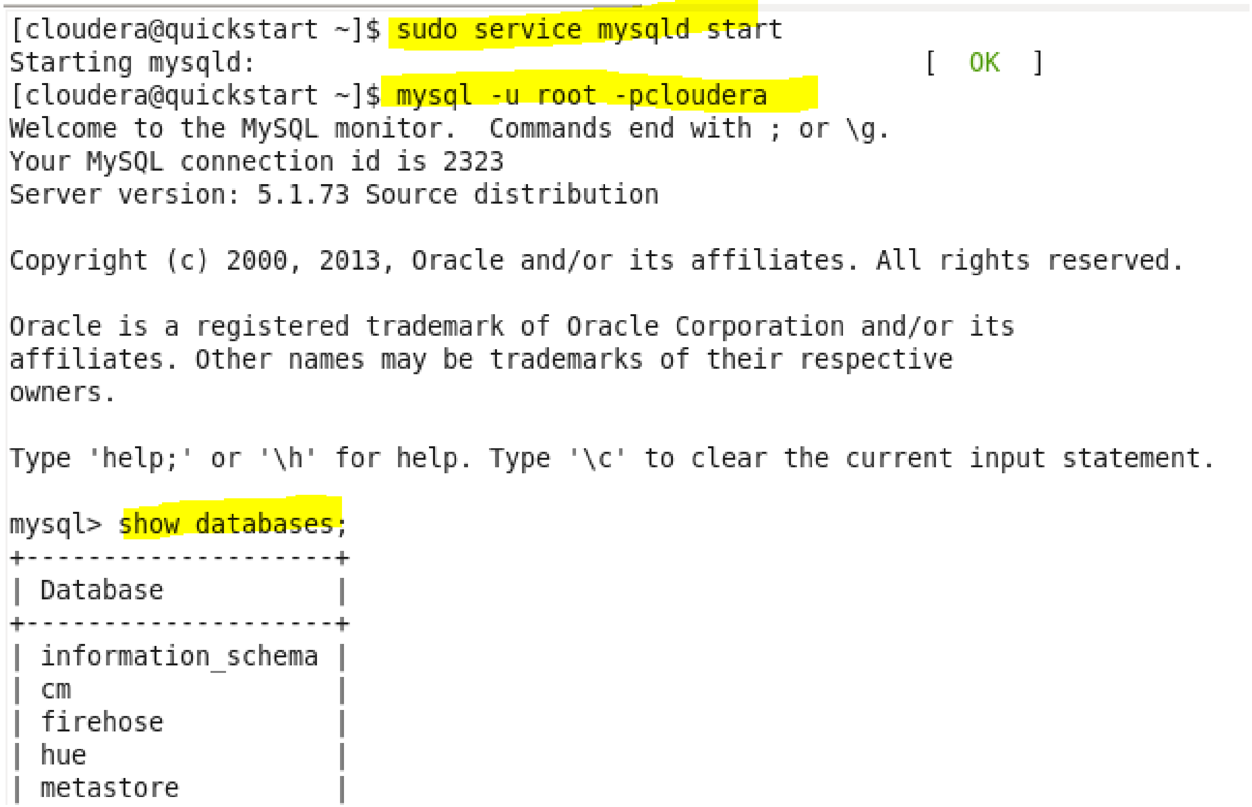
What I learned in ICP:

I could have learned about Sqoop and its functionality to transfer structured data between relational database. The sqoop export command transfers each record in hadoop as rows in tables to relational database and are delimited as specified.In this ICP I could successfully import and export mysql to hdfs by using sqoop and created hive tables through hql script and use sqoop to import and export tables. And also could get some knowledge about hql to implement some functionality.

ICP description what was the task you were performing and Screen shots that shows the successful execution of each required step of your code

<Task1>

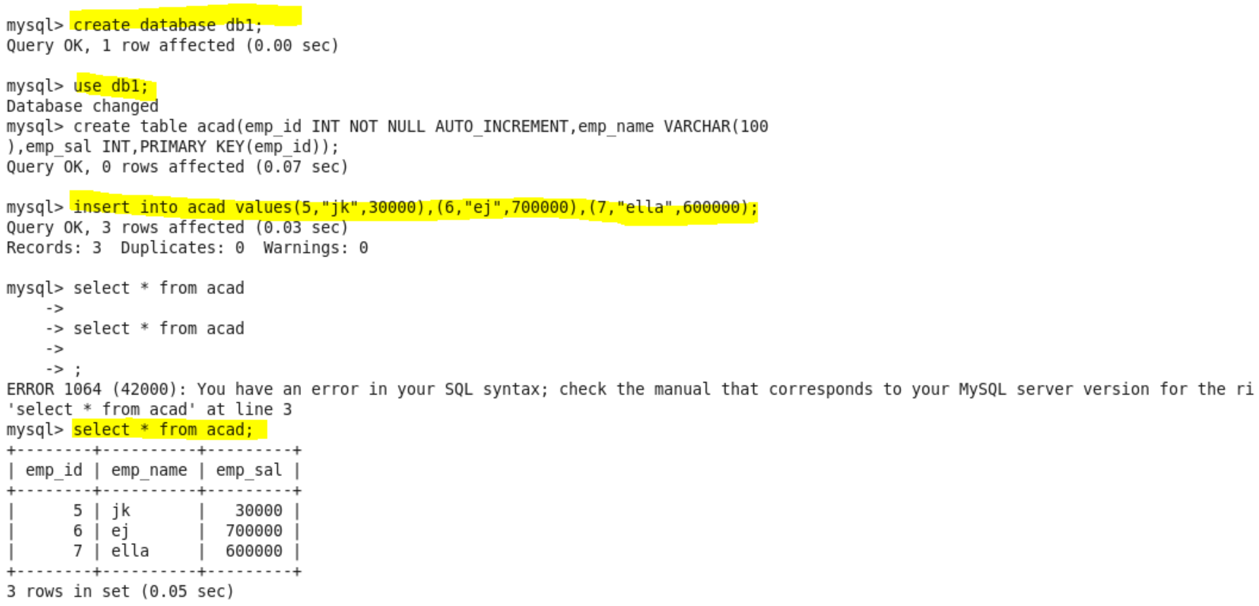
Use Sqoop to import and export MySQL Tables to HDFS.



```
[cloudera@quickstart ~]$ sudo service mysqld start
Starting mysqld:                                           [  OK  ]
[cloudera@quickstart ~]$ mysql -u root -pcloudera
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2323
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| cm                 |
| firehose           |
| hue                |
| metastore          |
```

First to start mysql service: ***sudo service mysqld start***

then enter mysql shell using: ***mysql -u root -pcloudera***



```
mysql> create database db1;
Query OK, 1 row affected (0.00 sec)

mysql> use db1;
Database changed
mysql> create table acad(emp_id INT NOT NULL AUTO_INCREMENT,emp_name VARCHAR(100
),emp_sal INT,PRIMARY KEY(emp_id));
Query OK, 0 rows affected (0.07 sec)

mysql> insert into acad values(5,"jk",30000),(6,"ej",700000),(7,"ella",600000);
Query OK, 3 rows affected (0.03 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> select * from acad
    ->
    -> select * from acad
    ->
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the ri
'select * from acad' at line 3
mysql> select * from acad;
+--------+----------+---------+
| emp_id | emp_name | emp_sal |
+--------+----------+---------+
|      5 | jk       |   30000 |
|      6 | ej       |  700000 |
|      7 | ella     |  600000 |
+--------+----------+---------+
3 rows in set (0.05 sec)
```

Command to create a new database: create database db1;

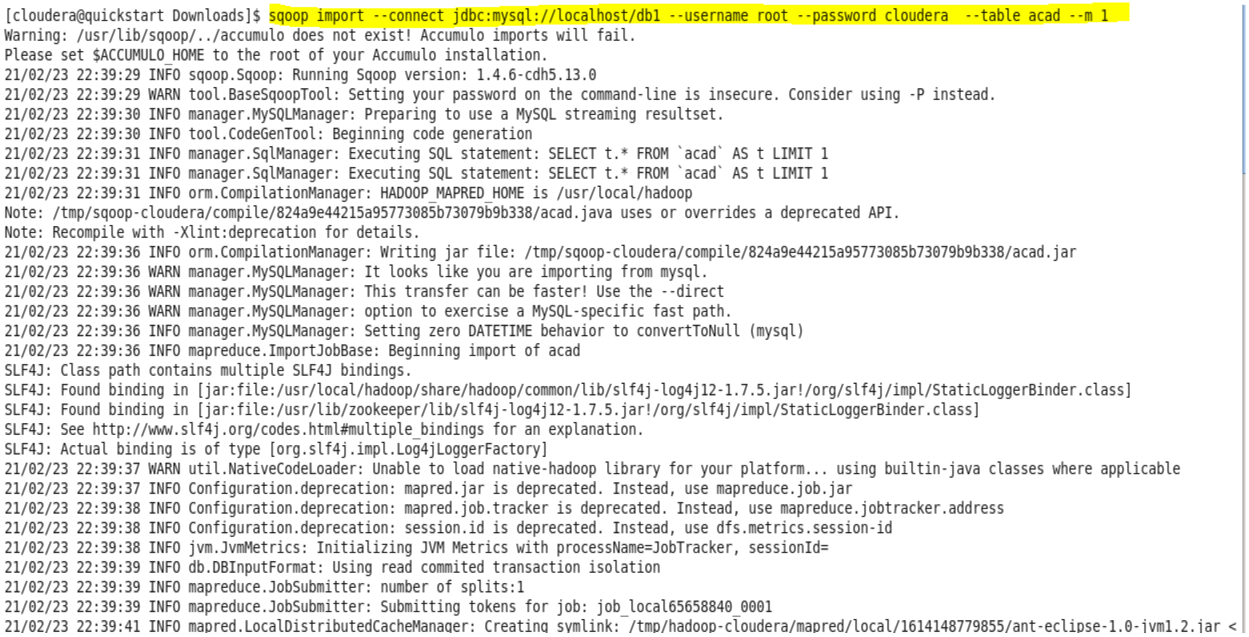
Command for using the database: use db1;

Created a table called acad with its attributes: create table acad(emp_id INT NOT NULL AUTO_INCREMENT,emp_name VARCHAR(100),emp_sal INT,PRIMARY KEY(emp_id)

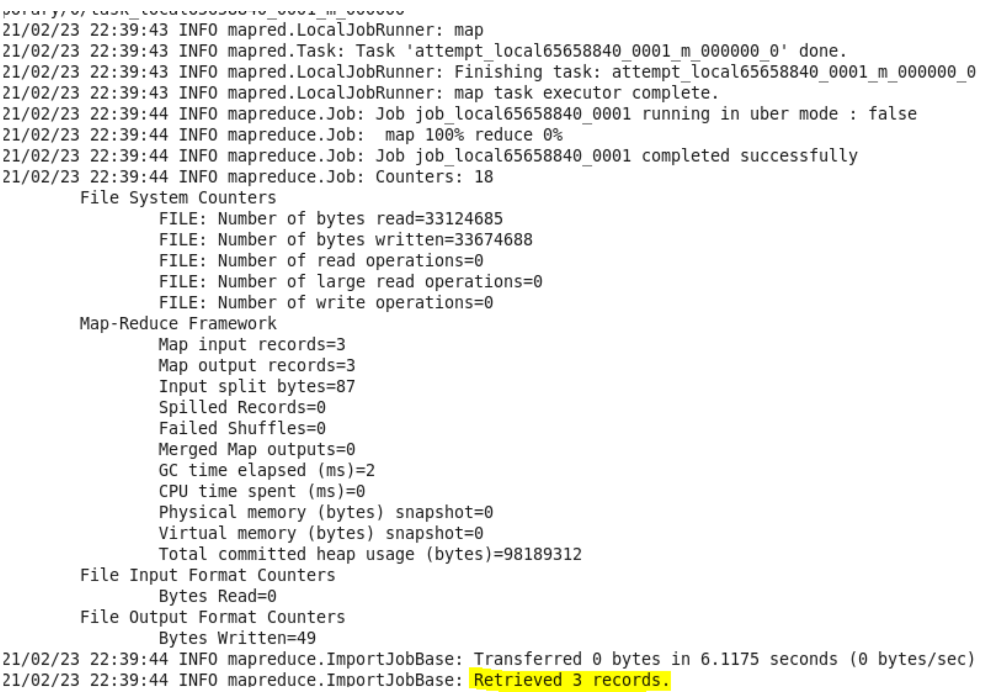
Inserting three data to the table: insert into acad values(5,”jk”,30000), (6,”ej”, 700000),(7,”ella”,600000);

Showing all the data in the table acad: select * from acad;

1)Importing a table

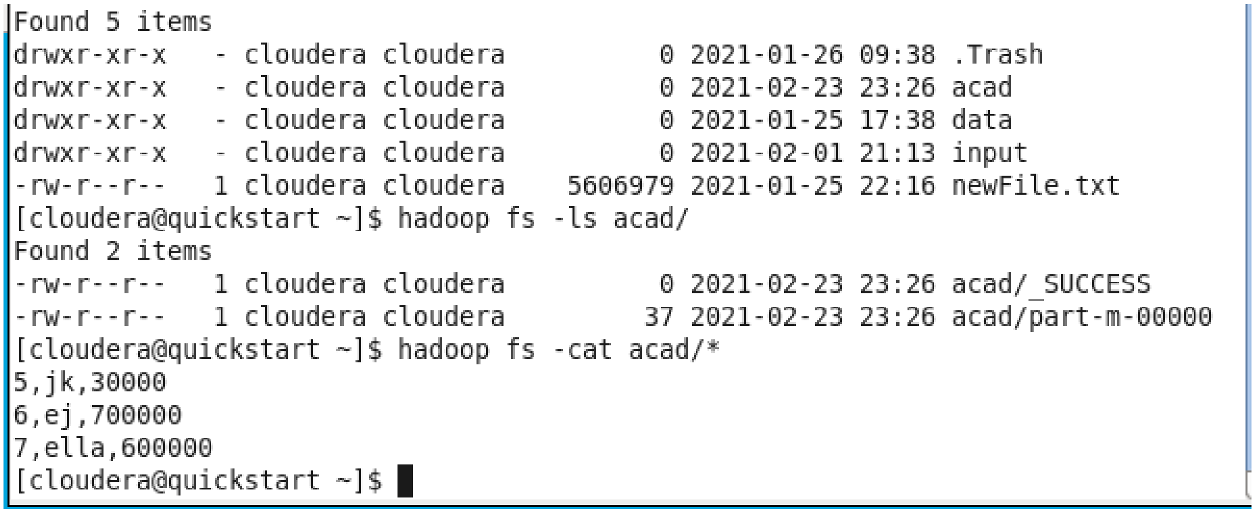


```
[cloudera@quickstart Downloads]$ sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera  --table acad --m 1
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
21/02/23 22:39:29 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
21/02/23 22:39:29 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
21/02/23 22:39:30 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
21/02/23 22:39:30 INFO tool.CodeGenTool: Beginning code generation
21/02/23 22:39:31 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `acad` AS t LIMIT 1
21/02/23 22:39:31 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `acad` AS t LIMIT 1
21/02/23 22:39:31 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop
Note: /tmp/sqoop-cloudera/compile/824a9e44215a95773085b73079b9b338/acad.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
21/02/23 22:39:36 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/824a9e44215a95773085b73079b9b338/acad.jar
21/02/23 22:39:36 WARN manager.MySQLManager: It looks like you are importing from mysql.
21/02/23 22:39:36 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
21/02/23 22:39:36 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
21/02/23 22:39:36 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
21/02/23 22:39:36 INFO mapreduce.ImportJobBase: Beginning import of acad
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/zookeeper/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
21/02/23 22:39:37 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
21/02/23 22:39:37 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
21/02/23 22:39:38 INFO Configuration.deprecation: mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
21/02/23 22:39:38 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
21/02/23 22:39:38 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
21/02/23 22:39:39 INFO db.DBInputFormat: Using read commited transaction isolation
21/02/23 22:39:39 INFO mapreduce.JobSubmitter: number of splits:1
21/02/23 22:39:39 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local65658840_0001
21/02/23 22:39:41 INFO mapred.LocalDistributedCacheManager: Creating symlink: /tmp/hadoop-cloudera/mapred/local/1614148779855/ant-eclipse-1.0-jvm1.2.jar <
```



```
porary/0/task_local65658840_0001_m_000000
21/02/23 22:39:43 INFO mapred.LocalJobRunner: map
21/02/23 22:39:43 INFO mapred.Task: Task 'attempt_local65658840_0001_m_000000_0' done.
21/02/23 22:39:43 INFO mapred.LocalJobRunner: Finishing task: attempt_local65658840_0001_m_000000_0
21/02/23 22:39:43 INFO mapred.LocalJobRunner: map task executor complete.
21/02/23 22:39:44 INFO mapreduce.Job: Job job_local65658840_0001 running in uber mode : false
21/02/23 22:39:44 INFO mapreduce.Job:  map 100% reduce 0%
21/02/23 22:39:44 INFO mapreduce.Job: Job job_local65658840_0001 completed successfully
21/02/23 22:39:44 INFO mapreduce.Job: Counters: 18
        File System Counters
                FILE: Number of bytes read=33124685
                FILE: Number of bytes written=33674688
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
        Map-Reduce Framework
                Map input records=3
                Map output records=3
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=2
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=98189312
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=49
21/02/23 22:39:44 INFO mapreduce.ImportJobBase: Transferred 0 bytes in 6.1175 seconds (0 bytes/sec)
21/02/23 22:39:44 INFO mapreduce.ImportJobBase: Retrieved 3 records.
```

Import a table: sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table acad --m 1



```
Found 5 items
drwxr-xr-x   - cloudera cloudera          0 2021-01-26 09:38 .Trash
drwxr-xr-x   - cloudera cloudera          0 2021-02-23 23:26 acad
drwxr-xr-x   - cloudera cloudera          0 2021-01-25 17:38 data
drwxr-xr-x   - cloudera cloudera          0 2021-02-01 21:13 input
-rw-r--r--   1 cloudera cloudera    5606979 2021-01-25 22:16 newFile.txt
[cloudera@quickstart ~]$ hadoop fs -ls acad/
Found 2 items
-rw-r--r--   1 cloudera cloudera          0 2021-02-23 23:26 acad/_SUCCESS
-rw-r--r--   1 cloudera cloudera         37 2021-02-23 23:26 acad/part-m-00000
[cloudera@quickstart ~]$ hadoop fs -cat acad/*
5,jk,30000
6,ej,700000
7,ella,600000
[cloudera@quickstart ~]$ █
```

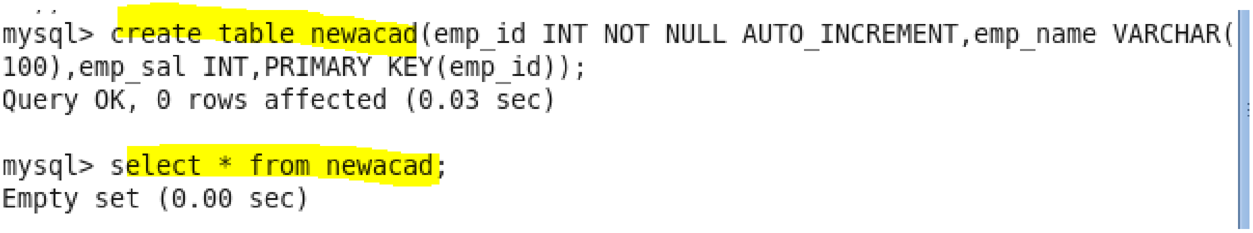
Showing the result of the importing in hdfs:

hadoop fs -ls

hadoop fs -ls acad/

hadoop fs -cat acad/*

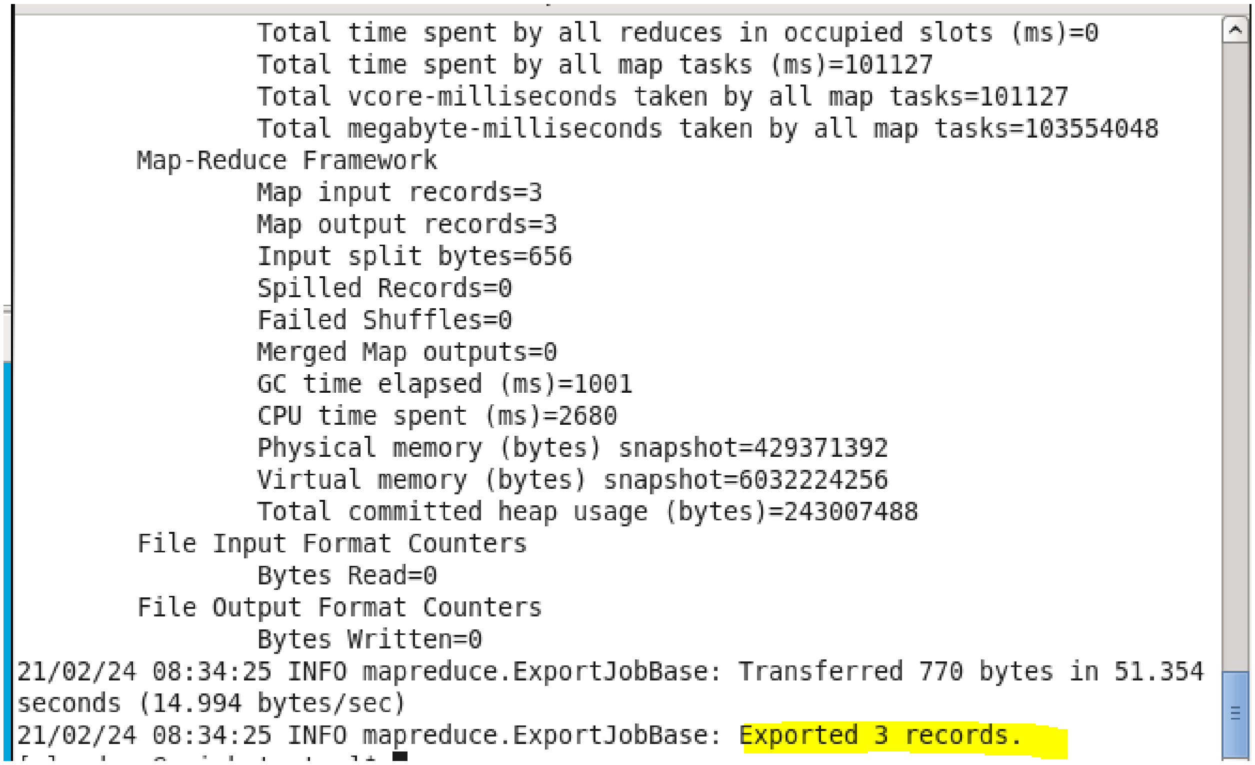
2)Exporting the table



```
mysql> create table newacad(emp_id INT NOT NULL AUTO_INCREMENT,emp_name VARCHAR(
100),emp_sal INT,PRIMARY KEY(emp_id));
Query OK, 0 rows affected (0.03 sec)

mysql> select * from newacad;
Empty set (0.00 sec)
```

Create a new table in mysql

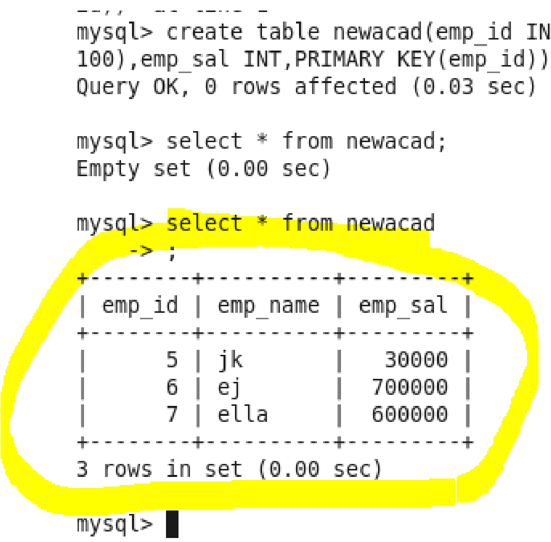
create table newacad(emp_id INT NOT NULL AUTO_INCREMENT, emp_name VARCHAR(100), emp_sal INT, PRIMARY KEY(emp_id));



```
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=101127
                Total vcore-milliseconds taken by all map tasks=101127
                Total megabyte-milliseconds taken by all map tasks=103554048
        Map-Reduce Framework
                Map input records=3
                Map output records=3
                Input split bytes=656
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=1001
                CPU time spent (ms)=2680
                Physical memory (bytes) snapshot=429371392
                Virtual memory (bytes) snapshot=6032224256
                Total committed heap usage (bytes)=243007488
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
21/02/24 08:34:25 INFO mapreduce.ExportJobBase: Transferred 770 bytes in 51.354
seconds (14.994 bytes/sec)
21/02/24 08:34:25 INFO mapreduce.ExportJobBase: Exported 3 records.
```

Exporting table from hdfs to newacad table:

sqoop export --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table newacad --export-dir /user/cloudera/ acad/part-m-00000

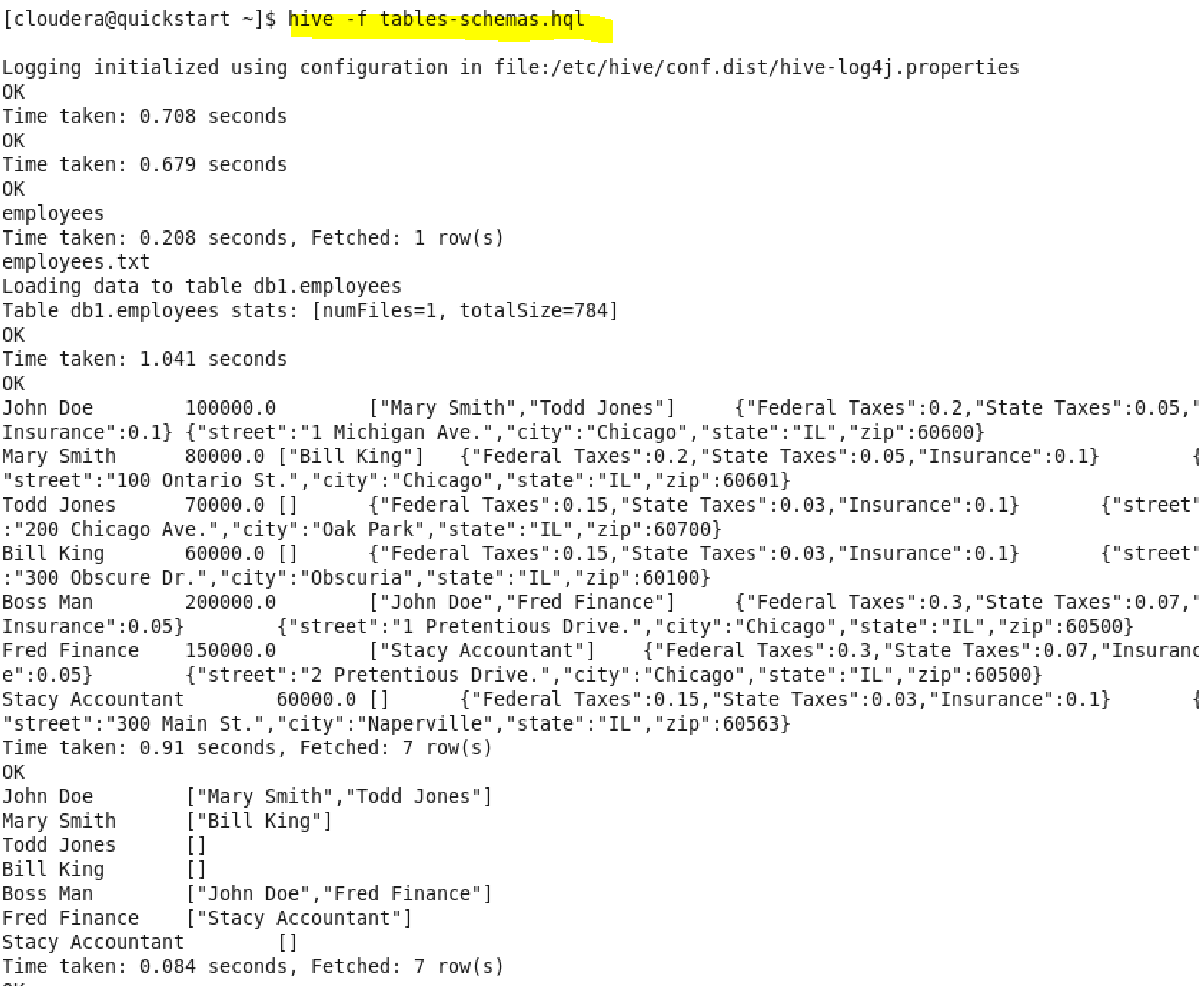


```
mysql> create table newacad(emp_id INT NOT NULL AUTO_INCREMENT,emp_name VARCHAR(
100),emp_sal INT,PRIMARY KEY(emp_id));
Query OK, 0 rows affected (0.03 sec)

mysql> select * from newacad;
Empty set (0.00 sec)

mysql> select * from newacad
    -> ;
+--------+----------+---------+
| emp_id | emp_name | emp_sal |
+--------+----------+---------+
|      5 | jk       |   30000 |
|      6 | ej       |  700000 |
|      7 | ella     |  600000 |
+--------+----------+---------+
3 rows in set (0.00 sec)

mysql> █
```

One can find out that the table is successfully exported to the mysql

<Task2>

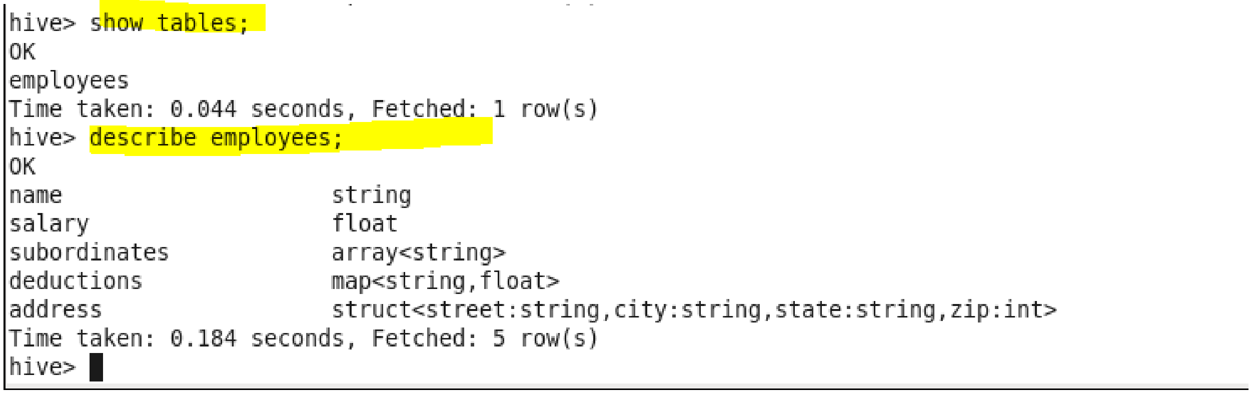
Create Hive Tables through HQL Script, Use Sqoop to import and export tables to Relational Databases



```
[cloudera@quickstart ~]$ hive -f tables-schemas.hql

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.properties
OK
Time taken: 0.708 seconds
OK
Time taken: 0.679 seconds
OK
employees
Time taken: 0.208 seconds, Fetched: 1 row(s)
employees.txt
Loading data to table db1.employees
Table db1.employees stats: [numFiles=1, totalSize=784]
OK
Time taken: 1.041 seconds
OK
John Doe        100000.0        ["Mary Smith","Todd Jones"]     {"Federal Taxes":0.2,"State Taxes":0.05,"
Insurance":0.1} {"street":"1 Michigan Ave.","city":"Chicago","state":"IL","zip":60600}
Mary Smith      80000.0 ["Bill King"]   {"Federal Taxes":0.2,"State Taxes":0.05,"Insurance":0.1}        {
"street":"100 Ontario St.","city":"Chicago","state":"IL","zip":60601}
Todd Jones      70000.0 []      {"Federal Taxes":0.15,"State Taxes":0.03,"Insurance":0.1}       {"street"
:"200 Chicago Ave.","city":"Oak Park","state":"IL","zip":60700}
Bill King       60000.0 []      {"Federal Taxes":0.15,"State Taxes":0.03,"Insurance":0.1}       {"street"
:"300 Obscure Dr.","city":"Obscuria","state":"IL","zip":60100}
Boss Man        200000.0        ["John Doe","Fred Finance"]     {"Federal Taxes":0.3,"State Taxes":0.07,"
Insurance":0.05}        {"street":"1 Pretentious Drive.","city":"Chicago","state":"IL","zip":60500}
Fred Finance    150000.0        ["Stacy Accountant"]    {"Federal Taxes":0.3,"State Taxes":0.07,"Insuranc
e":0.05}        {"street":"2 Pretentious Drive.","city":"Chicago","state":"IL","zip":60500}
Stacy Accountant        60000.0 []      {"Federal Taxes":0.15,"State Taxes":0.03,"Insurance":0.1}       {
"street":"300 Main St.","city":"Naperville","state":"IL","zip":60563}
Time taken: 0.91 seconds, Fetched: 7 row(s)
OK
John Doe        ["Mary Smith","Todd Jones"]
Mary Smith      ["Bill King"]
Todd Jones      []
Bill King       []
Boss Man        ["John Doe","Fred Finance"]
Fred Finance    ["Stacy Accountant"]
Stacy Accountant        []
Time taken: 0.084 seconds, Fetched: 7 row(s)
OK
```

1)Modified tables-schemas.hql and Table and Schema Creation through HQL Script

hive –f tables-schema.hql

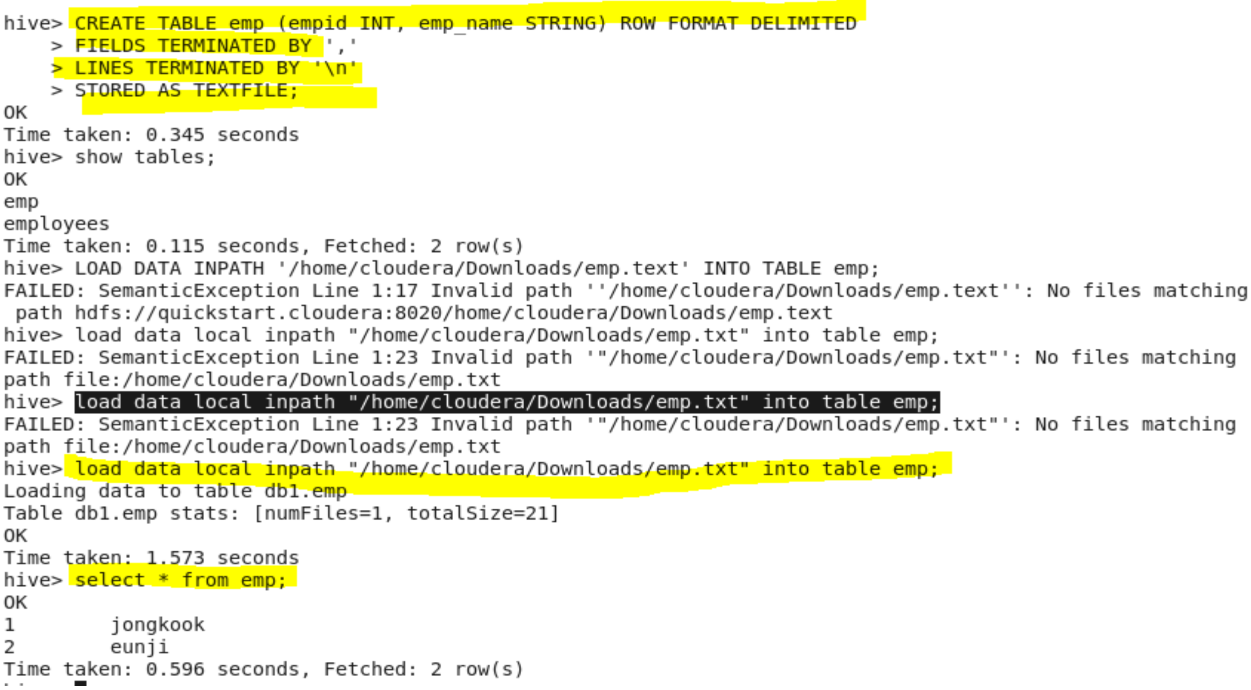


```
hive> show tables;
OK
employees
Time taken: 0.044 seconds, Fetched: 1 row(s)
hive> describe employees;
OK
name                    string
salary                  float
subordinates            array<string>
deductions              map<string,float>
address                 struct<street:string,city:string,state:string,zip:int>
Time taken: 0.184 seconds, Fetched: 5 row(s)
hive> 
```

Thanks to the tables-schema.hql, one can find out that employees table is created.

show tables;

describe employees;

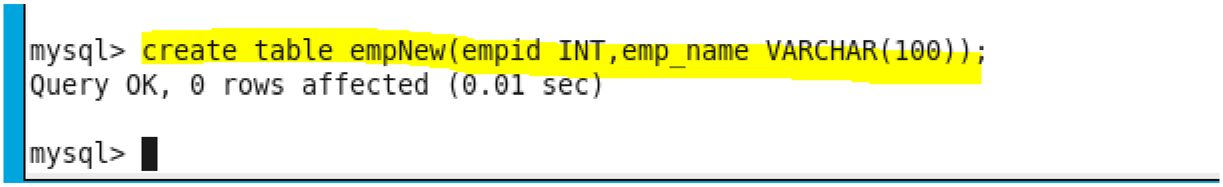


```
hive> CREATE TABLE emp (empid INT, emp_name STRING) ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY ','
    > LINES TERMINATED BY '\n'
    > STORED AS TEXTFILE;
OK
Time taken: 0.345 seconds
hive> show tables;
OK
emp
employees
Time taken: 0.115 seconds, Fetched: 2 row(s)
hive> LOAD DATA INPATH '/home/cloudera/Downloads/emp.text' INTO TABLE emp;
FAILED: SemanticException Line 1:17 Invalid path ''/home/cloudera/Downloads/emp.text'': No files matching
 path hdfs://quickstart.cloudera:8020/home/cloudera/Downloads/emp.text
hive> load data local inpath "/home/cloudera/Downloads/emp.txt" into table emp;
FAILED: SemanticException Line 1:23 Invalid path '"/home/cloudera/Downloads/emp.txt"': No files matching
path file:/home/cloudera/Downloads/emp.txt
hive> load data local inpath "/home/cloudera/Downloads/emp.txt" into table emp;
FAILED: SemanticException Line 1:23 Invalid path '"/home/cloudera/Downloads/emp.txt"': No files matching
path file:/home/cloudera/Downloads/emp.txt
hive> load data local inpath "/home/cloudera/Downloads/emp.txt" into table emp;
Loading data to table db1.emp
Table db1.emp stats: [numFiles=1, totalSize=21]
OK
Time taken: 1.573 seconds
hive> select * from emp;
OK
1       jongkook
2       eunji
Time taken: 0.596 seconds, Fetched: 2 row(s)
```

Created table named emp in hive and load the data file from the local

create table emp(empid INT, emp_name STRING) row format delimited fields terminated by "," stored as textfile;

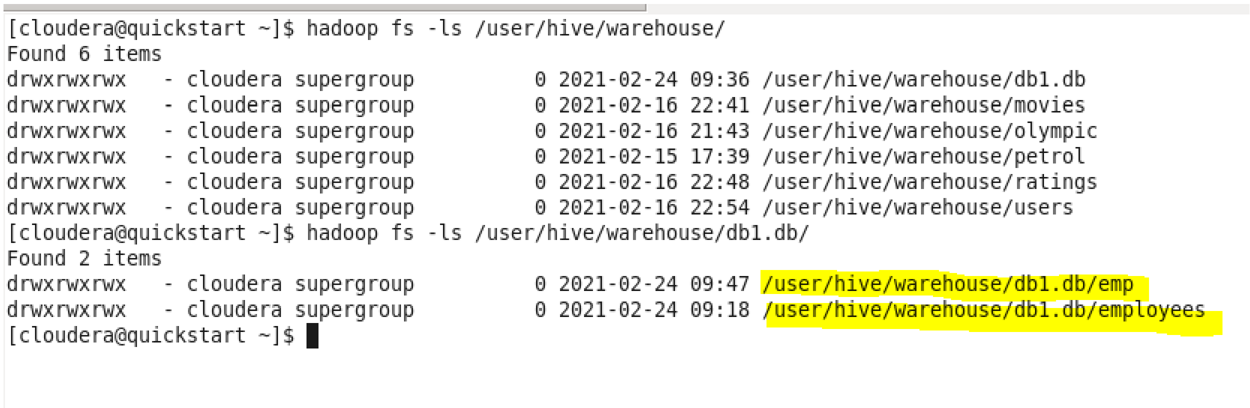
load data local inpath "/home/cloudera/Downloads/emp.txt" into table emp;



```
mysql> create table empNew(empid INT,emp_name VARCHAR(100));
Query OK, 0 rows affected (0.01 sec)

mysql> 
```

In my sql, I created empty table named empNew

create table empNew(empid INT, emp_name VARCHAR(100));



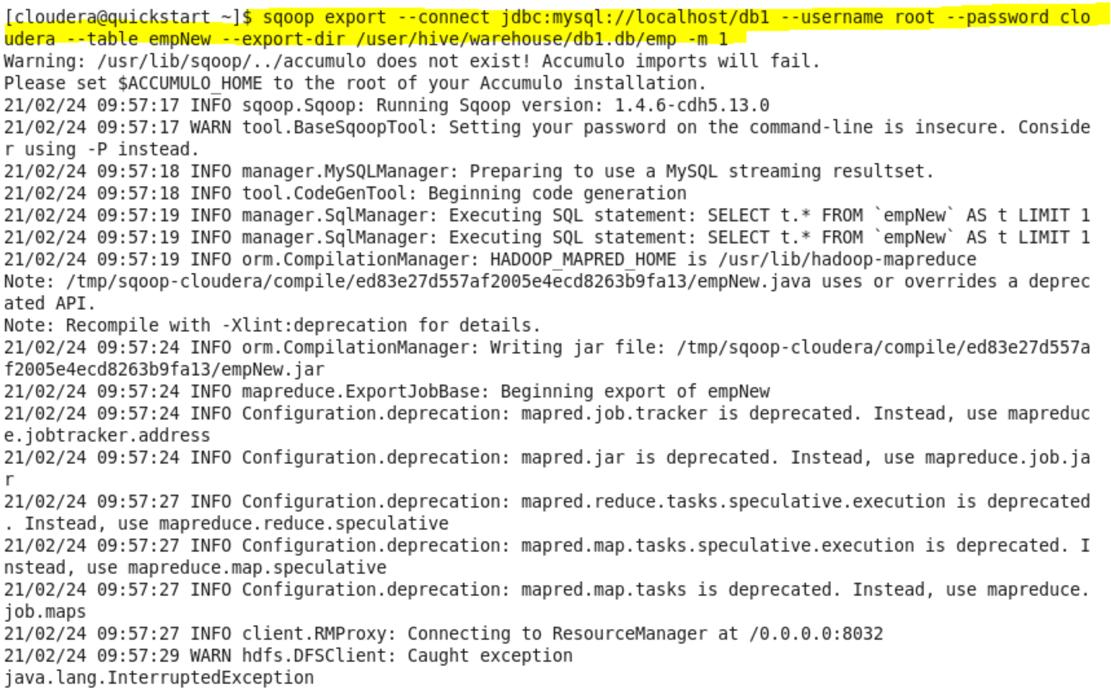
```
[cloudera@quickstart ~]$ hadoop fs -ls /user/hive/warehouse/
Found 6 items
drwxrwxrwx   - cloudera supergroup          0 2021-02-24 09:36 /user/hive/warehouse/db1.db
drwxrwxrwx   - cloudera supergroup          0 2021-02-16 22:41 /user/hive/warehouse/movies
drwxrwxrwx   - cloudera supergroup          0 2021-02-16 21:43 /user/hive/warehouse/olympic
drwxrwxrwx   - cloudera supergroup          0 2021-02-15 17:39 /user/hive/warehouse/petrol
drwxrwxrwx   - cloudera supergroup          0 2021-02-16 22:48 /user/hive/warehouse/ratings
drwxrwxrwx   - cloudera supergroup          0 2021-02-16 22:54 /user/hive/warehouse/users
[cloudera@quickstart ~]$ hadoop fs -ls /user/hive/warehouse/db1.db/
Found 2 items
drwxrwxrwx   - cloudera supergroup          0 2021-02-24 09:47 /user/hive/warehouse/db1.db/emp
drwxrwxrwx   - cloudera supergroup          0 2021-02-24 09:18 /user/hive/warehouse/db1.db/employees
[cloudera@quickstart ~]$ 
```

Then find the location of hive tables in hdfs

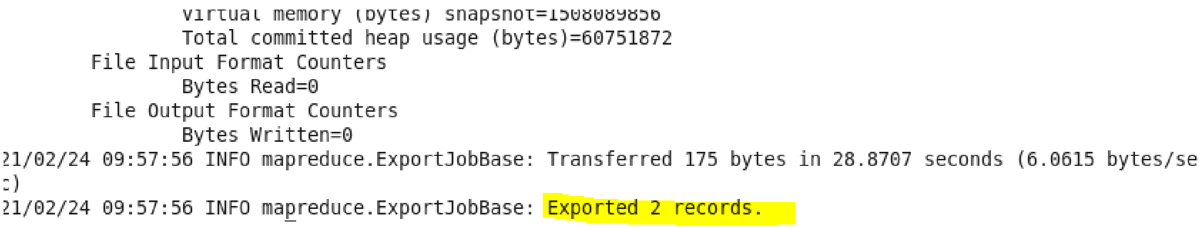
hadoop fs -ls /user/hive/warehouse/

hadoop fs -ls /user/hive/warehouse/db1.db/

2)Exporting Table to MySQL empNew through sqoop

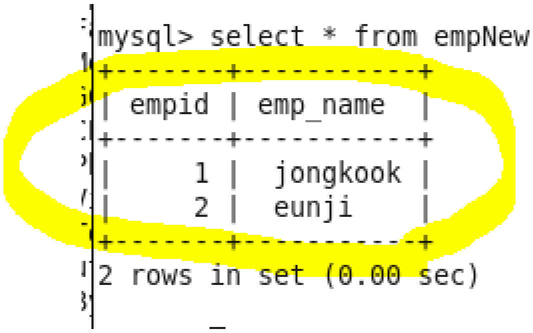


```
[cloudera@quickstart ~]$ sqoop export --connect jdbc:mysql://localhost/db1 --username root --password clo
udera --table empNew --export-dir /user/hive/warehouse/db1.db/emp -m 1
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
21/02/24 09:57:17 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
21/02/24 09:57:17 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Conside
r using -P instead.
21/02/24 09:57:18 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
21/02/24 09:57:18 INFO tool.CodeGenTool: Beginning code generation
21/02/24 09:57:19 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `empNew` AS t LIMIT 1
21/02/24 09:57:19 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `empNew` AS t LIMIT 1
21/02/24 09:57:19 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-cloudera/compile/ed83e27d557af2005e4ecd8263b9fa13/empNew.java uses or overrides a deprec
ated API.
Note: Recompile with -Xlint:deprecation for details.
21/02/24 09:57:24 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/ed83e27d557a
f2005e4ecd8263b9fa13/empNew.jar
21/02/24 09:57:24 INFO mapreduce.ExportJobBase: Beginning export of empNew
21/02/24 09:57:24 INFO Configuration.deprecation: mapred.job.tracker is deprecated. Instead, use mapreduc
e.jobtracker.address
21/02/24 09:57:24 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.ja
r
21/02/24 09:57:27 INFO Configuration.deprecation: mapred.reduce.tasks.speculative.execution is deprecated
. Instead, use mapreduce.reduce.speculative
21/02/24 09:57:27 INFO Configuration.deprecation: mapred.map.tasks.speculative.execution is deprecated. I
nstead, use mapreduce.map.speculative
21/02/24 09:57:27 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.
job.maps
21/02/24 09:57:27 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/02/24 09:57:29 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
```



```
                Virtual memory (bytes) snapshot=1508089856
                Total committed heap usage (bytes)=60751872
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
21/02/24 09:57:56 INFO mapreduce.ExportJobBase: Transferred 175 bytes in 28.8707 seconds (6.0615 bytes/se
c)
21/02/24 09:57:56 INFO mapreduce.ExportJobBase: Exported 2 records.
```

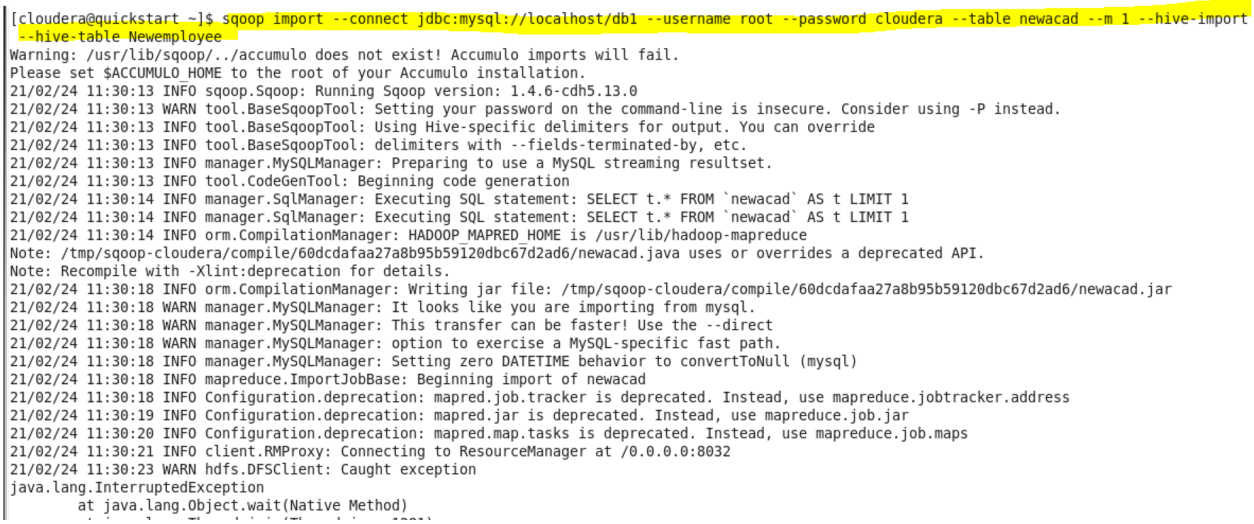
sqoop export --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table empNew --export-dir /user/hive/warehouse/db1.db/emp -m 1



```
mysql> select * from empNew;
+-------+-----------+
| empid | emp_name  |
+-------+-----------+
|     1 |  jongkook |
|     2 |  eunji    |
+-------+-----------+
2 rows in set (0.00 sec)
```

One can find out that table is successfully exported to the mysql

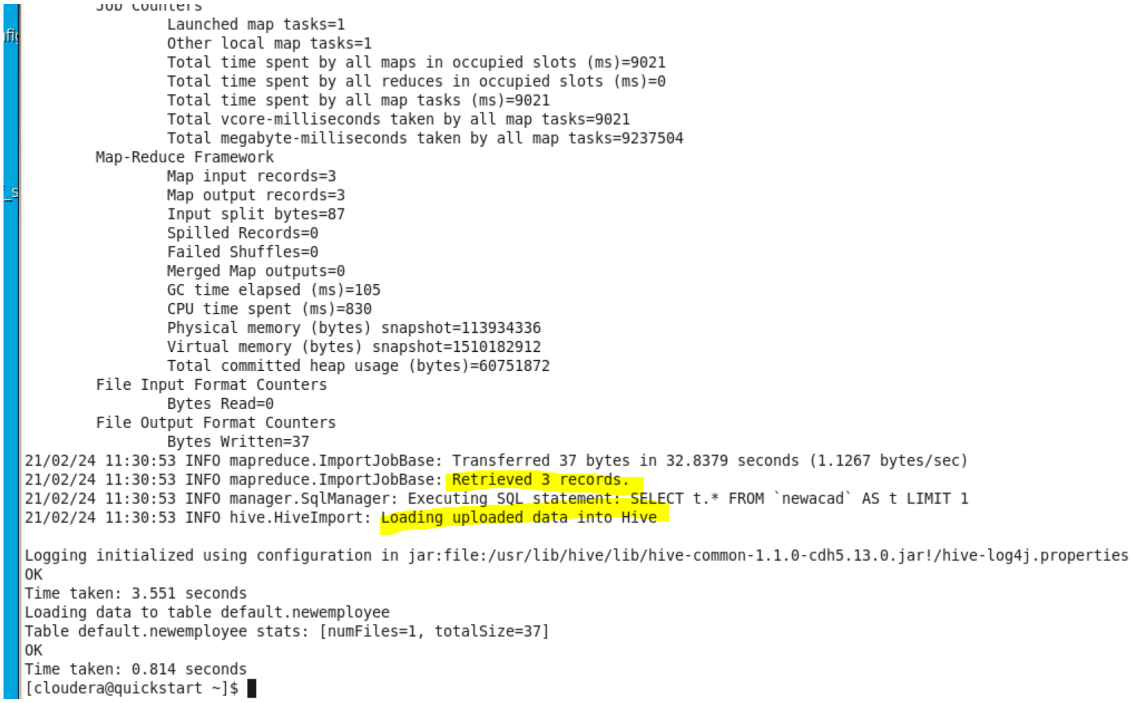
3)Importing from MySQL to Hive



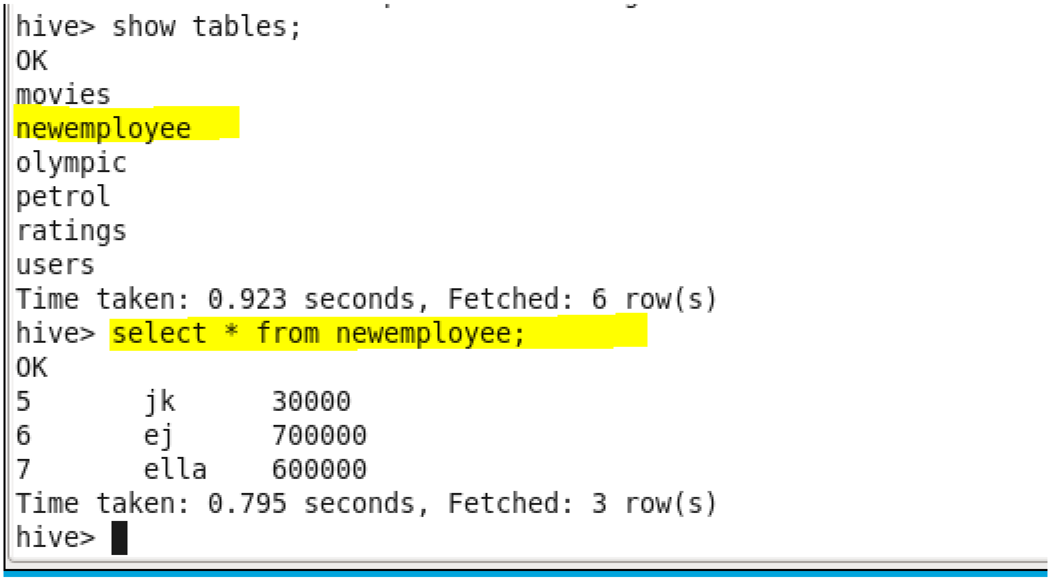
```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table newacad --m 1 --hive-import
 --hive-table Newemployee
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
21/02/24 11:30:13 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
21/02/24 11:30:13 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
21/02/24 11:30:13 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can override
21/02/24 11:30:13 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
21/02/24 11:30:13 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
21/02/24 11:30:13 INFO tool.CodeGenTool: Beginning code generation
21/02/24 11:30:14 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `newacad` AS t LIMIT 1
21/02/24 11:30:14 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `newacad` AS t LIMIT 1
21/02/24 11:30:14 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-cloudera/compile/60dcdafaa27a8b95b59120dbc67d2ad6/newacad.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
21/02/24 11:30:18 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/60dcdafaa27a8b95b59120dbc67d2ad6/newacad.jar
21/02/24 11:30:18 WARN manager.MySQLManager: It looks like you are importing from mysql.
21/02/24 11:30:18 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
21/02/24 11:30:18 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
21/02/24 11:30:18 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
21/02/24 11:30:18 INFO mapreduce.ImportJobBase: Beginning import of newacad
21/02/24 11:30:18 INFO Configuration.deprecation: mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
21/02/24 11:30:19 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
21/02/24 11:30:20 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
21/02/24 11:30:21 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/02/24 11:30:23 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
```

I already created table named newacad in mysql and I import that table named as Newemployee in hive:

sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table newacad --m 1 --hive-import --hive-table Newemployee



```
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=9021
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=9021
                Total vcore-milliseconds taken by all map tasks=9021
                Total megabyte-milliseconds taken by all map tasks=9237504
        Map-Reduce Framework
                Map input records=3
                Map output records=3
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=105
                CPU time spent (ms)=830
                Physical memory (bytes) snapshot=113934336
                Virtual memory (bytes) snapshot=1510182912
                Total committed heap usage (bytes)=60751872
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=37
21/02/24 11:30:53 INFO mapreduce.ImportJobBase: Transferred 37 bytes in 32.8379 seconds (1.1267 bytes/sec)
21/02/24 11:30:53 INFO mapreduce.ImportJobBase: Retrieved 3 records.
21/02/24 11:30:53 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `newacad` AS t LIMIT 1
21/02/24 11:30:53 INFO hive.HiveImport: Loading uploaded data into Hive

Logging initialized using configuration in jar:file:/usr/lib/hive/lib/hive-common-1.1.0-cdh5.13.0.jar!/hive-log4j.properties
OK
Time taken: 3.551 seconds
Loading data to table default.newemployee
Table default.newemployee stats: [numFiles=1, totalSize=37]
OK
Time taken: 0.814 seconds
[cloudera@quickstart ~]$ █
```



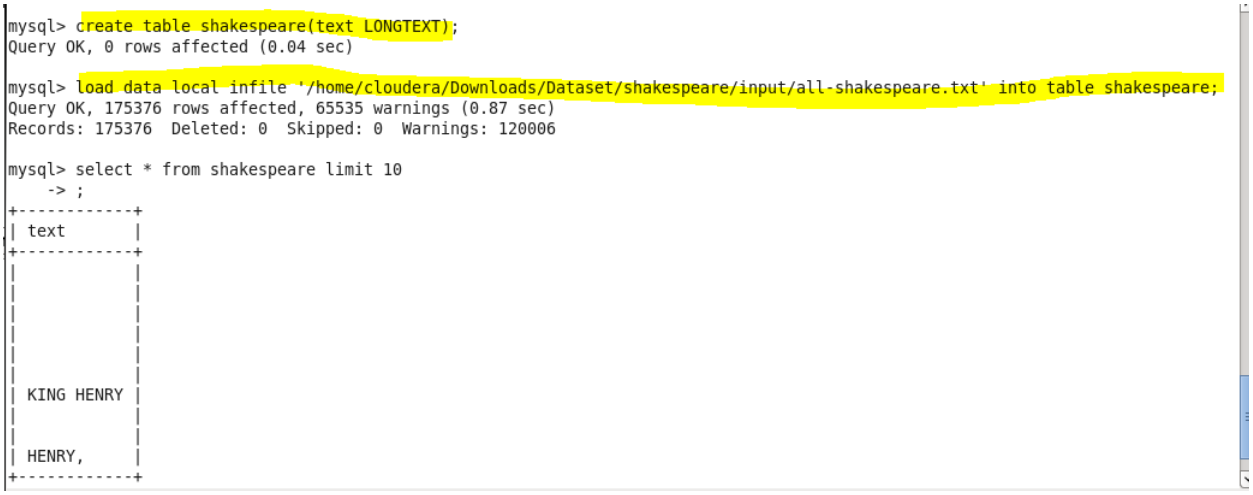
```
hive> show tables;
OK
movies
newemployee
olympic
petrol
ratings
users
Time taken: 0.923 seconds, Fetched: 6 row(s)
hive> select * from newemployee;
OK
5       jk      30000
6       ej      700000
7       ella    600000
Time taken: 0.795 seconds, Fetched: 3 row(s)
hive> 
```

One can find out that the table is successfully imported in the hive

<Task3>

Perform three queries from databases (Statistics, WordCount, and Identifying pattern)

1)Create SQL Table and import as Hive

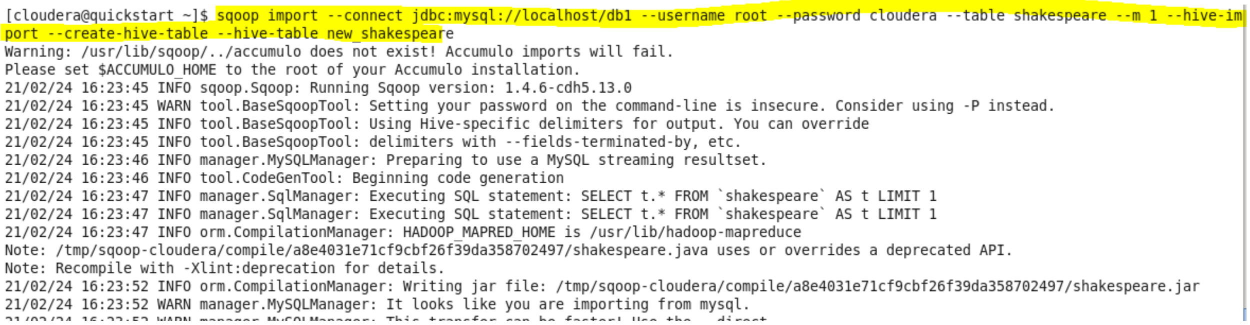


```
mysql> create table shakespeare(text LONGTEXT);
Query OK, 0 rows affected (0.04 sec)

mysql> load data local infile '/home/cloudera/Downloads/Dataset/shakespeare/input/all-shakespeare.txt' into table shakespeare;
Query OK, 175376 rows affected, 65535 warnings (0.87 sec)
Records: 175376  Deleted: 0  Skipped: 0  Warnings: 120006

mysql> select * from shakespeare limit 10
    -> ;
+------------+
| text       |
+------------+
|            |
|            |
|            |
|            |
|            |
|            |
| KING HENRY |
|            |
|            |
| HENRY,     |
+------------+
```

I created new table named Shakespeare in mysql and load the Shakespeare text file into it

create table shakespeare(text LONGTEXT);

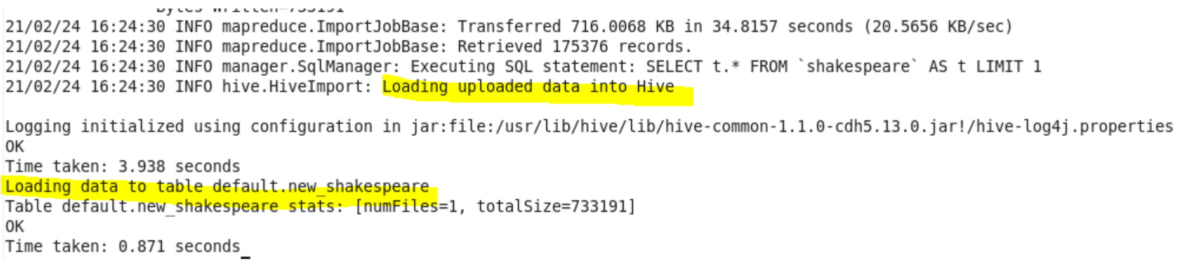
load data local infile '/home/cloudera/Downloads/Dataset/shakespeare/input/all-shakespeare.txt' into table article;



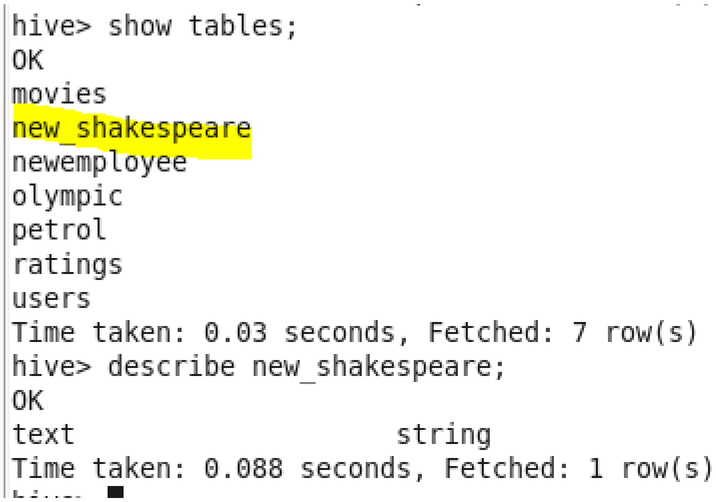
```
[cloudera@quickstart ~]$ sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table shakespeare --m 1 --hive-im
port --create-hive-table --hive-table new_shakespeare
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
21/02/24 16:23:45 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
21/02/24 16:23:45 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
21/02/24 16:23:45 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can override
21/02/24 16:23:45 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
21/02/24 16:23:46 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
21/02/24 16:23:46 INFO tool.CodeGenTool: Beginning code generation
21/02/24 16:23:47 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `shakespeare` AS t LIMIT 1
21/02/24 16:23:47 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `shakespeare` AS t LIMIT 1
21/02/24 16:23:47 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-cloudera/compile/a8e4031e71cf9cbf26f39da358702497/shakespeare.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
21/02/24 16:23:52 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/a8e4031e71cf9cbf26f39da358702497/shakespeare.jar
21/02/24 16:23:52 WARN manager.MySQLManager: It looks like you are importing from mysql.
21/02/24 16:23:52 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
```

Importing the table from my sql to hive as named new_shakespeare

sqoop import --connect jdbc:mysql://localhost/db1 --username root --password cloudera --table shakespeare --m 1 --hive-import --create-hive-table --hive-table new_shakespeare



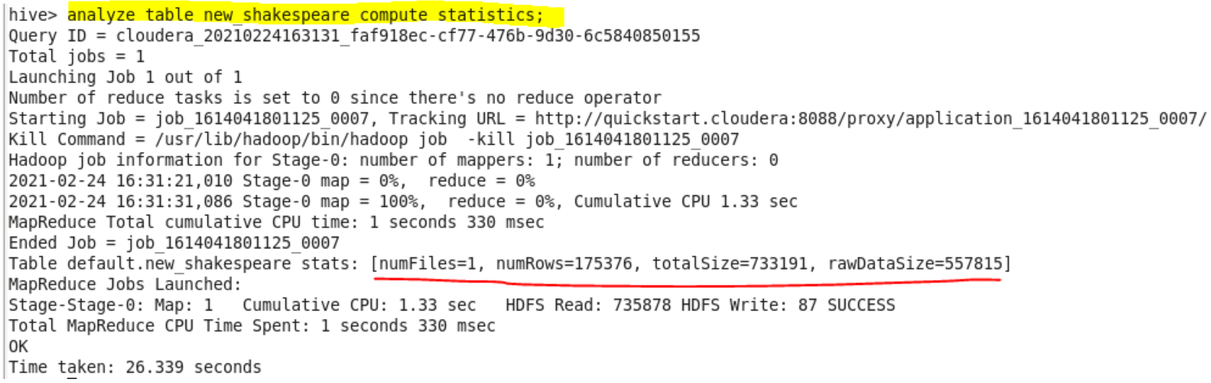
```
                Bytes Written=733191
21/02/24 16:24:30 INFO mapreduce.ImportJobBase: Transferred 716.0068 KB in 34.8157 seconds (20.5656 KB/sec)
21/02/24 16:24:30 INFO mapreduce.ImportJobBase: Retrieved 175376 records.
21/02/24 16:24:30 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `shakespeare` AS t LIMIT 1
21/02/24 16:24:30 INFO hive.HiveImport: Loading uploaded data into Hive

Logging initialized using configuration in jar:file:/usr/lib/hive/lib/hive-common-1.1.0-cdh5.13.0.jar!/hive-log4j.properties
OK
Time taken: 3.938 seconds
Loading data to table default.new_shakespeare
Table default.new_shakespeare stats: [numFiles=1, totalSize=733191]
OK
Time taken: 0.871 seconds_
```



```
hive> show tables;
OK
movies
new_shakespeare
newemployee
olympic
petrol
ratings
users
Time taken: 0.03 seconds, Fetched: 7 row(s)
hive> describe new_shakespeare;
OK
text                    string
Time taken: 0.088 seconds, Fetched: 1 row(s)
hive>
```

One can find out that the table is successfully imported in the hive

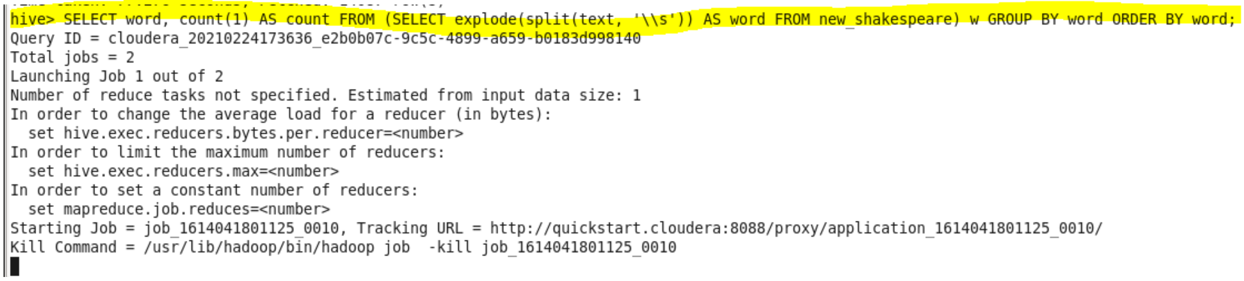
2)Queries of statistics



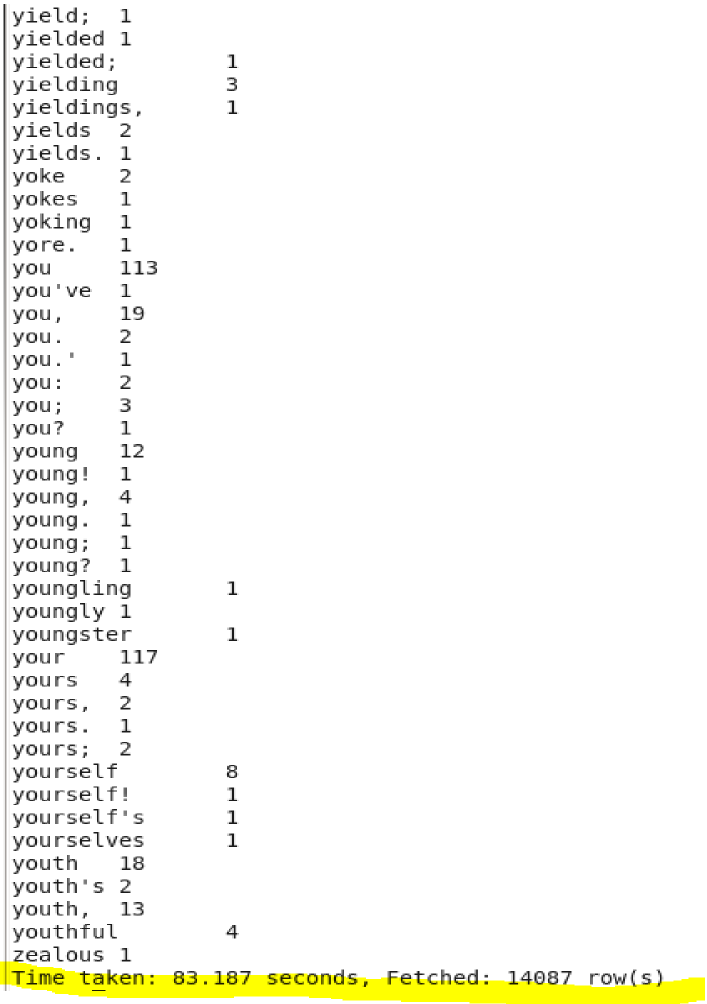
```
hive> analyze table new_shakespeare compute statistics;
Query ID = cloudera_20210224163131_faf918ec-cf77-476b-9d30-6c5840850155
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1614041801125_0007, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1614041801125_0007/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1614041801125_0007
Hadoop job information for Stage-0: number of mappers: 1; number of reducers: 0
2021-02-24 16:31:21,010 Stage-0 map = 0%,  reduce = 0%
2021-02-24 16:31:31,086 Stage-0 map = 100%,  reduce = 0%, Cumulative CPU 1.33 sec
MapReduce Total cumulative CPU time: 1 seconds 330 msec
Ended Job = job_1614041801125_0007
Table default.new_shakespeare stats: [numFiles=1, numRows=175376, totalSize=733191, rawDataSize=557815]
MapReduce Jobs Launched:
Stage-Stage-0: Map: 1   Cumulative CPU: 1.33 sec   HDFS Read: 735878 HDFS Write: 87 SUCCESS
Total MapReduce CPU Time Spent: 1 seconds 330 msec
OK
Time taken: 26.339 seconds
```

For existing tables and/or partitions, the user can issue the ANALYZE command to gather statistics and write them into Hive MetaStore. If one may or may not specify the partition specs. If the user doesn't specify any partition specs, statistics are gathered for the table as well as all the partitions

3)Queries of wordcount



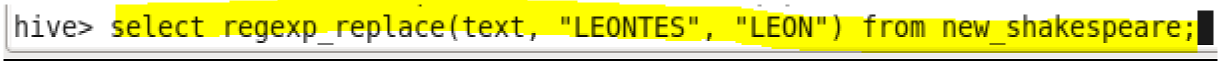
```
hive> SELECT word, count(1) AS count FROM (SELECT explode(split(text, '\\s')) AS word FROM new_shakespeare) w GROUP BY word ORDER BY word;
Query ID = cloudera_20210224173636_e2b0b07c-9c5c-4899-a659-b0183d998140
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated 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_1614041801125_0010, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1614041801125_0010/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1614041801125_0010
```



```
yield;  1
yielded 1
yielded;        1
yielding        3
yieldings,      1
yields  2
yields. 1
yoke    2
yokes   1
yoking  1
yore.   1
you     113
you've  1
you,    19
you.    2
you.'   1
you:    2
you;    3
you?    1
young   12
young!  1
young,  4
young.  1
young;  1
young?  1
youngling       1
youngly 1
youngster       1
your    117
yours   4
yours,  2
yours.  1
yours;  2
yourself        8
yourself!       1
yourself's      1
yourselves      1
youth   18
youth's 2
youth,  13
youthful        4
zealous 1
Time taken: 83.187 seconds, Fetched: 14087 row(s)
```

SELECT word, count(1) AS count FROM (SELECT explode(split(text, '\\s')) AS word FROM new_shakespeare) w GROUP BY word ORDER BY word;

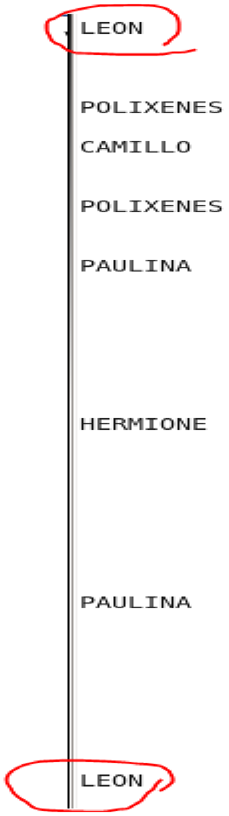
4)Queries of Identifying pattern



```
hive> select regexp_replace(text, "LEONTES", "LEON") from new_shakespeare;
```

I used regex_replace for identifying patter queries. It Returns the string resulting from replacing all substrings in INITIAL_STRING that match the java regular expression syntax defined in PATTERN with instances of REPLACEMENT.

select regexp_replace(text, “LEONTES”, “LEON”) from new_shakespeare;



```
LEON


POLIXENES
CAMILLO

POLIXENES

PAULINA




HERMIONE





PAULINA






LEON
```

One can find out that LEONTIS is replaced with LEON

References:

https://cwiki.apache.org/confluence/display/Hive/StatsDev
https://stackoverflow.com/questions/10039949/word-count-program-in-hive
https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF
https://www.tutorialspoint.com/sqoop/sqoop_installation.htm
https://www.tutorialspoint.com/hive/hive_create_database.htm