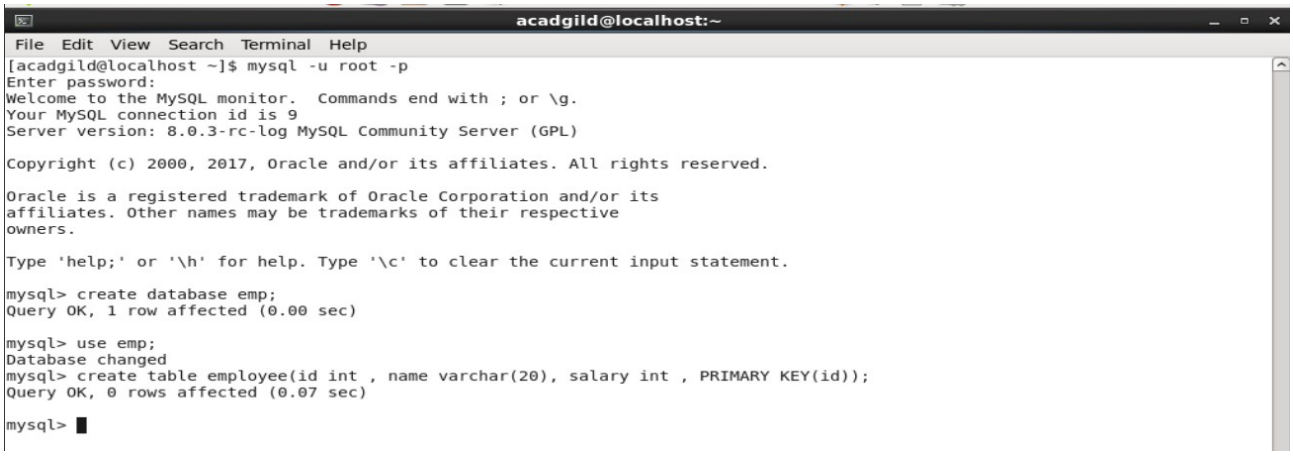
Bigdata Assignment – 4.3

Perform and explain the code flow and the associated result for the below tasks. Candidates should
create and use their own employee dataset for the same. Share the screenshot of the commands used
and its associated result.
● Transfer data between Mysql and HDFS (Import and Export) using Sqoop.
● Transfer data between Mysql and Hive (Import and Export only selected columns) using Sqoop.

Task 1 -

For importing data into hdfs from mysql

- Created a database and used it and it created a table.
  create database emp;
  use emp;
  create table employee(id int, name varchar(20) , salary int , PRIMARY KEY(id));



- Inserted data into the table.
  Insert into employee values(1,'Ritesh',20000);

```
mysql> insert into employee  values(1,'Ritesh',20000);
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee  values(2,'Shyam',30000);
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee  values(3,'Ram',10000);
Query OK, 1 row affected (0.00 sec)

mysql> select * from employee;
+----+--------+--------+
| id | name   | salary |
+----+--------+--------+
|  1 | Ritesh |  20000 |
|  2 | Shyam  |  30000 |
|  3 | Ram    |  10000 |
+----+--------+--------+
3 rows in set (0.00 sec)

mysql>
```

- Imported the mysql data into hdfs by using sqoop
  sqoop import –connect jdbc:mysql://localhost/emp –username root –password Root@123 –table employee –m 1 –target -dir /sqoopomport/emp

```
[acadgild@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/emp --username root --password Root@123 --table employe
e --m 1 --target-dir /sqoopomport/emp
```

- Then we checked whether the file is imported into hdfs or not, then its contents.
  hadoop fs -cat /sqoopomport/emp/part-m-00000

```
acadgild@localhost:~
[acadgild@localhost ~]$ hadoop fs -ls /sqoopomport
18/06/13 21:03:11 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 1 items
drwxr-xr-x   - acadgild supergroup          0 2018-06-13 21:01 /sqoopomport/emp
[acadgild@localhost ~]$ hadoop fs -ls /sqoopomport/emp
18/06/13 21:03:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2018-06-13 21:01 /sqoopomport/emp/_SUCCESS
-rw-r--r--   1 acadgild supergroup         41 2018-06-13 21:01 /sqoopomport/emp/part-m-00000
[acadgild@localhost ~]$ hadoop fs -cat /sqoopomport/emp/part-m-00000
18/06/13 21:04:08 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java cl
asses where applicable
1,Ritesh,20000
2,Shyam,30000
3,Ram,10000
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
```

In the above screenshot it is visible that the same data we got in the hdfs

For exporting data into mysql from hdfs:

- Used the created database and created a table in it with same columns.
  use emp;
  create table sqoop_export(id int, name varchar(20) , salary int , PRIMARY KEY(id));

```
mysql> use emp;
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> create table sqoop_export(id int , name varchar(20), salary int , PRIMARY KEY(id));
Query OK, 0 rows affected (0.03 sec)
```

- Transfered the data from hdfs into mysql using sqoop
  sqoop export –connect jdbd:mysql://localhost/emp –username 'root' –password Root@123 –table 'sqoop_export' –export-dir '/sqoopomport/emp/part-m-00000' –input-fields-terminated-by ',' -m 1 –columns id,name,salary

- Checked the data of the table
  select * from sqoop_export ;

```
mysql> select * from sqoop_export;
+----+--------+--------+
| id | name   | salary |
+----+--------+--------+
|  1 | Ritesh |  20000 |
|  2 | Shyam  |  30000 |
|  3 | Ram    |  10000 |
+----+--------+--------+
3 rows in set (0.00 sec)

mysql>
```

Task 2

Transfer of data from mysql into hive:

- Tranfer of data using sqoop from mysql table which was created earlier into hive.
  sqoop import –connect jdbc:mysql://localhost/emp –username 'root' –pasword Root@123 –split-by id –columns id,name,salary –table employee –target-dir 'sqoopimport/emphive' –hive-import –create-hivetable –hive-table default.nysqlemployee -m 1

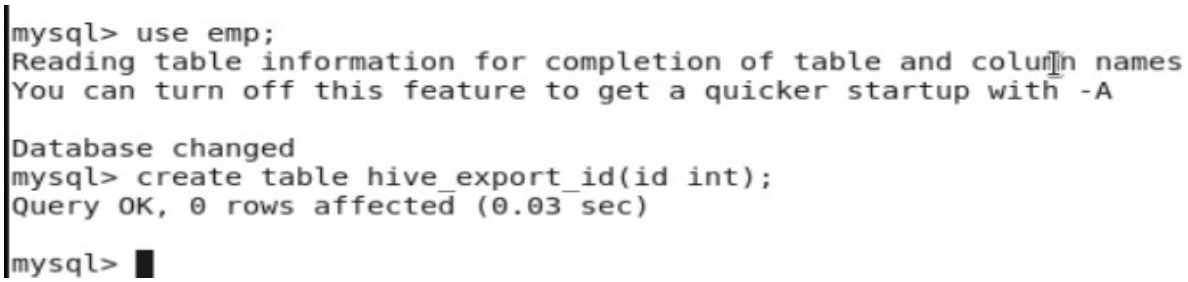
- In hive , in the default database we checked the table is present or not , then the content of the table.
  show databases;
  use default;
  show tables;
  select * from mysqlemployee;

In the above screenshot , we can see the contents of the table same as in mysql table.

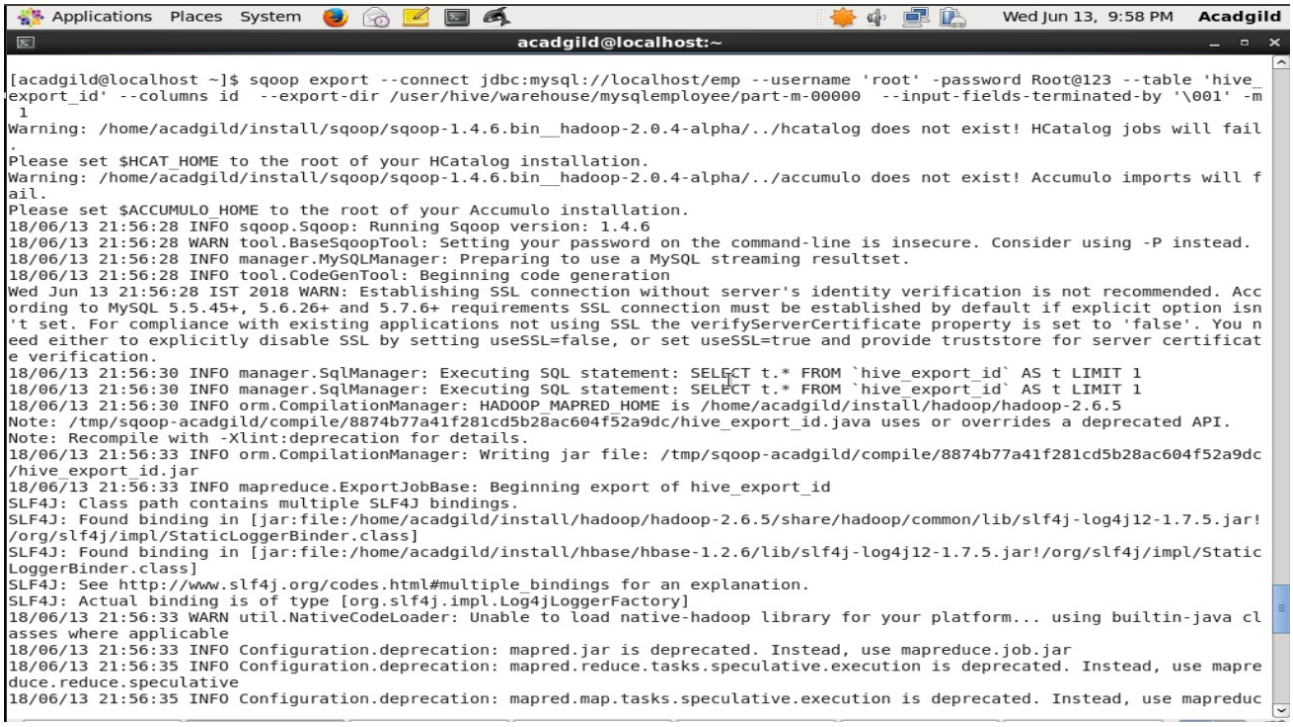
Transfer of data from hive into mysql:

- In mysql , in the emp database we created a table with id as only columns so tha we can export only id from hive.

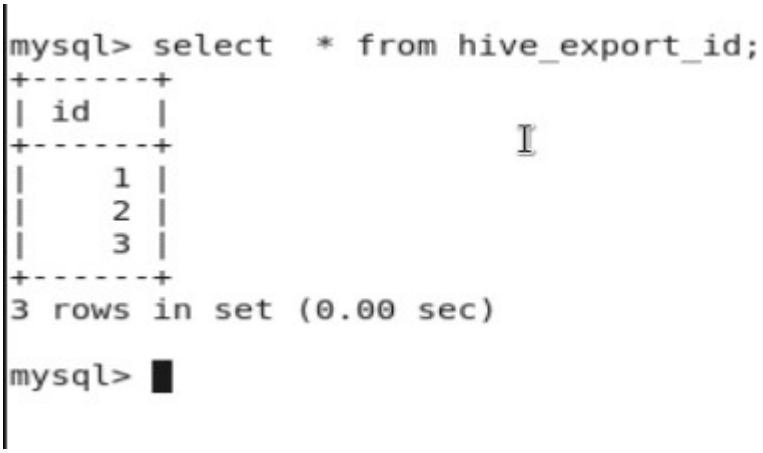
  use emp;
  create table hive_export_id(id int);



- We transfered the data from hive to mysql using sqoop.
  sqoop export –connect jdbc:mysql://localhost/emp –username 'root' –password Root@123 –table 'hive_export' –columns id –export-dir /user/hive/warehouse/mysqlemployee/part-m-00000 –input-fields-terminated-by '\001' -m 1



- In mysql , we checked the contents of the table where data is exported.
  select * from hive_export_id;



We can see we exported only id's from the hive table into mysql table.It is same as in the hive table.