Assignment 20.3

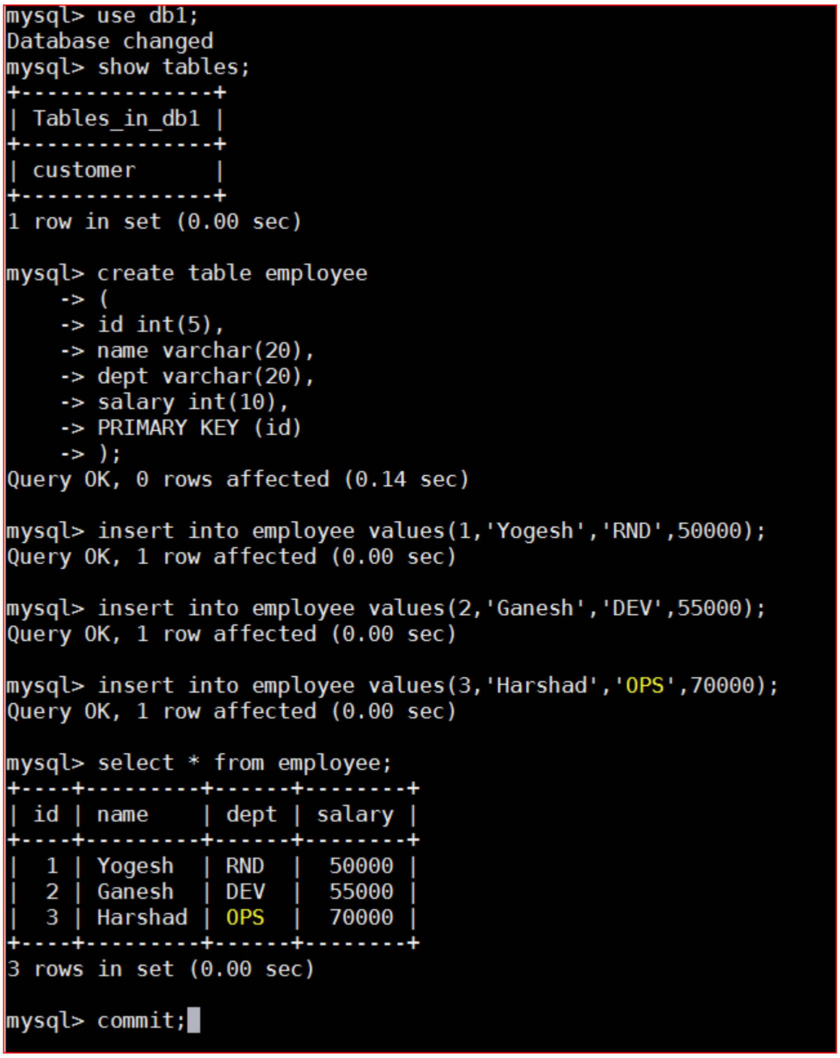
Problem Statement: Perform and explain the code flow and the associated result for the below tasks. Candidates should create and use their own employee dataset for the same. Share the screenshot of the commands used and its associated result.

Transfer data between Mysql and HDFS (Import and Export) using Sqoop.

1. Import:

Create table with below parameters in SQL:

```
use db1;

show tables;

create table employee

(

id int(5),

name varchar(20),

dept varchar(20),

salary int(10),

PRIMARY KEY (id)

);

insert into employee values(1,'Yogesh','RND',50000);

insert into employee values(2,'Ganesh','DEV',55000);

insert into employee values(3,'Harshad','OPS',70000);

select * from employee;

commit;
```



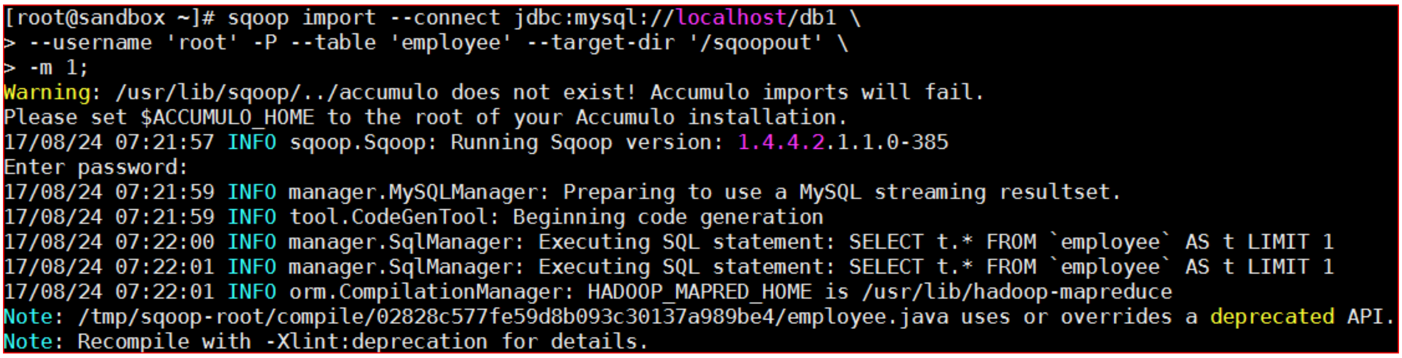
```
mysql> use db1;
Database changed
mysql> show tables;
+---------------+
| Tables_in_db1 |
+---------------+
| customer      |
+---------------+
1 row in set (0.00 sec)

mysql> create table employee
    -> (
    -> id int(5),
    -> name varchar(20),
    -> dept varchar(20),
    -> salary int(10),
    -> PRIMARY KEY (id)
    -> );
Query OK, 0 rows affected (0.14 sec)

mysql> insert into employee values(1,'Yogesh','RND',50000);
Query OK, 1 row affected (0.00 sec)

mysql> insert into employee values(2,'Ganesh','DEV',55000);
Query OK, 1 row affected (0.00 sec)

mysql> insert into employee values(3,'Harshad','OPS',70000);
Query OK, 1 row affected (0.00 sec)

mysql> select * from employee;
+----+---------+------+--------+
| id | name    | dept | salary |
+----+---------+------+--------+
|  1 | Yogesh  | RND  |  50000 |
|  2 | Ganesh  | DEV  |  55000 |
|  3 | Harshad | OPS  |  70000 |
+----+---------+------+--------+
3 rows in set (0.00 sec)

mysql> commit;
```

Run Sqoop command to import data into HDFS.

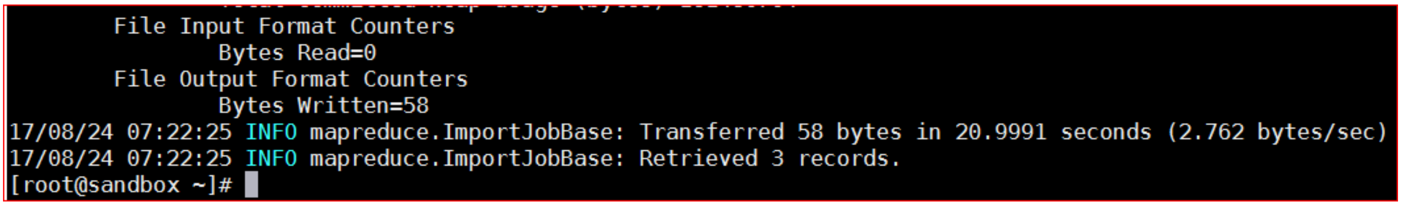
*sqoop import --connect jdbc:mysql://localhost/db1 *

*--username 'root' -P --table 'employee' --target-dir '/sqoopout' *

-m 1;



```
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'employee' --target-dir '/sqoopout' \
> -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.
17/08/24 07:21:57 INFO sqoop.Sqoop: Running Sqoop version: 1.4.4.2.1.1.0-385
Enter password:
17/08/24 07:21:59 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
17/08/24 07:21:59 INFO tool.CodeGenTool: Beginning code generation
17/08/24 07:22:00 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 07:22:01 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 07:22:01 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/02828c577fe59d8b093c30137a989be4/employee.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```



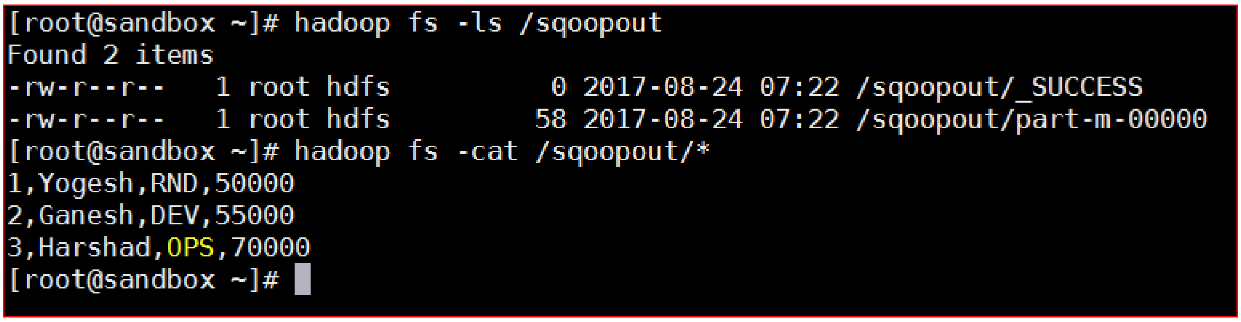
```
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=58
17/08/24 07:22:25 INFO mapreduce.ImportJobBase: Transferred 58 bytes in 20.9991 seconds (2.762 bytes/sec)
17/08/24 07:22:25 INFO mapreduce.ImportJobBase: Retrieved 3 records.
[root@sandbox ~]#
```

Data imported successfully.

Check data in the hdfs location /sqoopout

# hadoop fs -ls /sqoopout

*# hadoop fs -cat /sqoopout/**



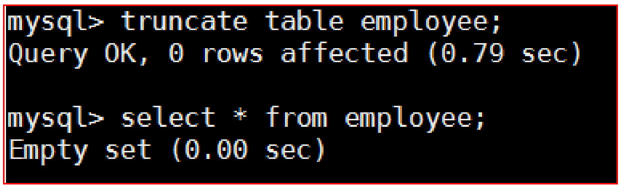
```
[root@sandbox ~]# hadoop fs -ls /sqoopout
Found 2 items
-rw-r--r--   1 root hdfs          0 2017-08-24 07:22 /sqoopout/_SUCCESS
-rw-r--r--   1 root hdfs         58 2017-08-24 07:22 /sqoopout/part-m-00000
[root@sandbox ~]# hadoop fs -cat /sqoopout/*
1,Yogesh,RND,50000
2,Ganesh,DEV,55000
3,Harshad,OPS,70000
[root@sandbox ~]# 
```

2. Export:

Truncate the table employee in SQL:

truncate table employee;

*select * from employee;*



```
mysql> truncate table employee;
Query OK, 0 rows affected (0.79 sec)

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

Run below sqoop command to load the data back to employee table in sql from the file generated by sqoop import in above command:

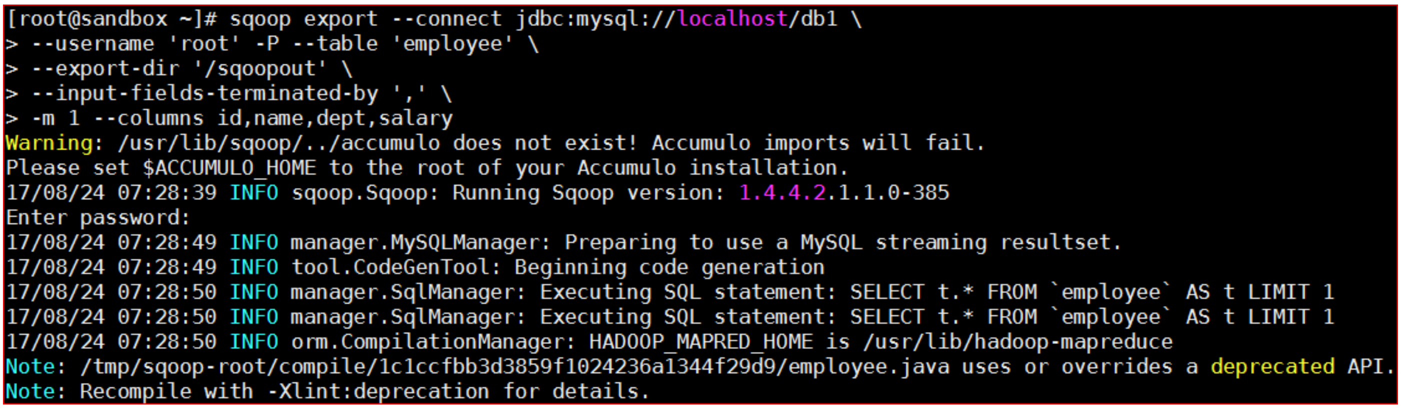
*sqoop export --connect jdbc:mysql://localhost/db1 *

*--username 'root' -P --table 'employee' *

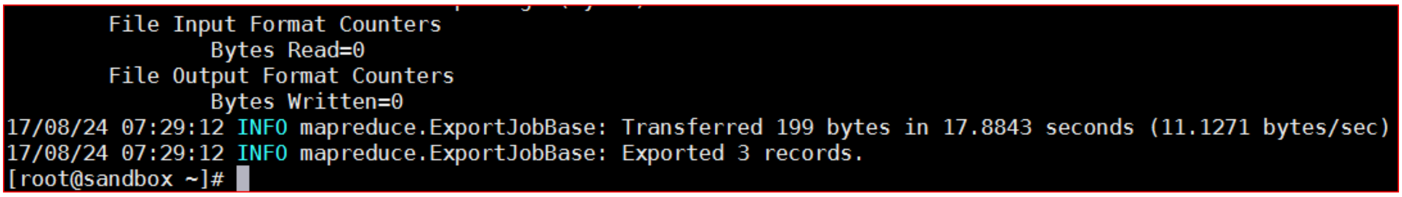
*--export-dir '/sqoopout' *

*--input-fields-terminated-by ',' *

-m 1 --columns id,name,dept,salary



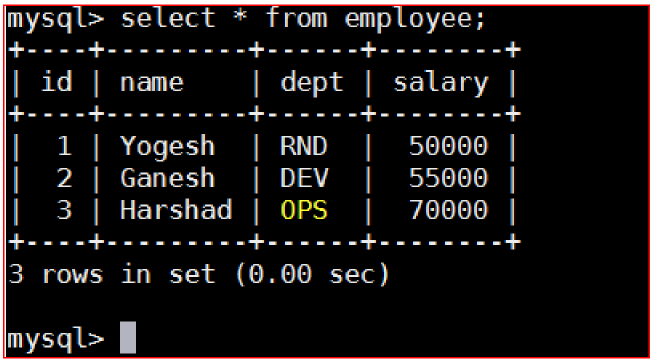
```
[root@sandbox ~]# sqoop export --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'employee' \
> --export-dir '/sqoopout' \
> --input-fields-terminated-by ',' \
> -m 1 --columns id,name,dept,salary
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
17/08/24 07:28:39 INFO sqoop.Sqoop: Running Sqoop version: 1.4.4.2.1.1.0-385
Enter password:
17/08/24 07:28:49 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
17/08/24 07:28:49 INFO tool.CodeGenTool: Beginning code generation
17/08/24 07:28:50 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 07:28:50 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 07:28:50 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/1c1ccfbb3d3859f1024236a1344f29d9/employee.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```



```
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
17/08/24 07:29:12 INFO mapreduce.ExportJobBase: Transferred 199 bytes in 17.8843 seconds (11.1271 bytes/sec)
17/08/24 07:29:12 INFO mapreduce.ExportJobBase: Exported 3 records.
[root@sandbox ~]#
```

Check the employee table in sql:

*select * from employee;*



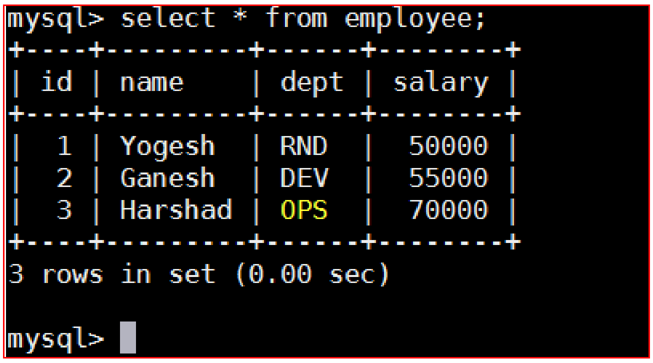
```
mysql> select * from employee;
+----+---------+------+--------+
| id | name    | dept | salary |
+----+---------+------+--------+
|  1 | Yogesh  | RND  |  50000 |
|  2 | Ganesh  | DEV  |  55000 |
|  3 | Harshad | OPS  |  70000 |
+----+---------+------+--------+
3 rows in set (0.00 sec)

mysql>
```

Transfer data between Mysql and Hive (Import and Export only selected columns) using Sqoop.

1. Import selected column:

Check employee table in sql:

*select * from employee;*



```
mysql> select * from employee;
+----+---------+------+--------+
| id | name    | dept | salary |
+----+---------+------+--------+
|  1 | Yogesh  | RND  |  50000 |
|  2 | Ganesh  | DEV  |  55000 |
|  3 | Harshad | OPS  |  70000 |
+----+---------+------+--------+
3 rows in set (0.00 sec)

mysql>
```

Run the below import command to select columns in the table employee.

*sqoop import --connect jdbc:mysql://localhost/db1 *

*--username 'root' -P --table 'employee' *

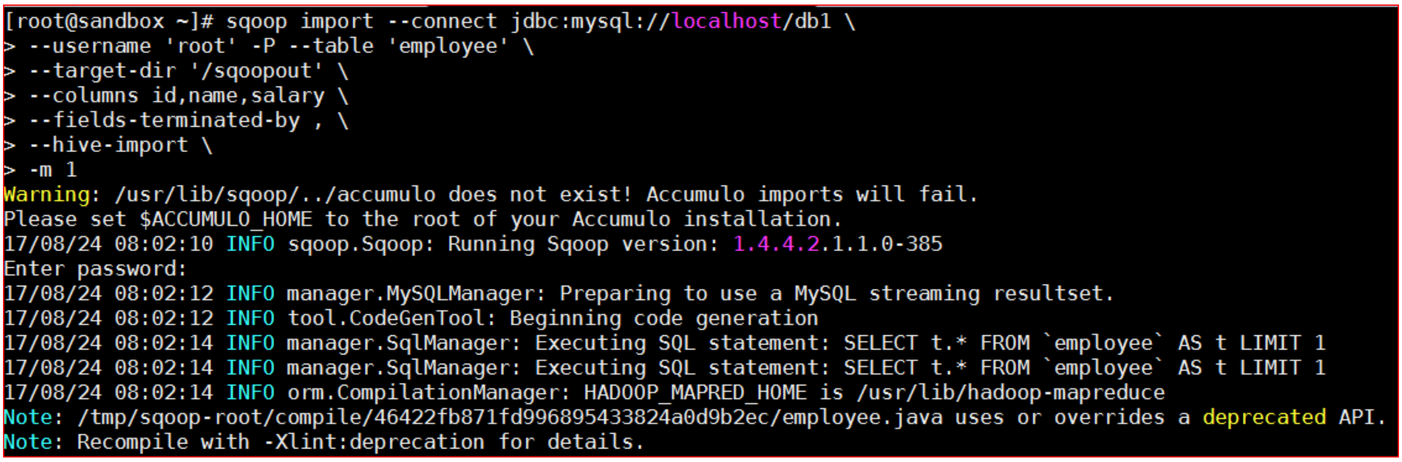
*--target-dir '/sqoopout' *

*--columns id,name,salary *

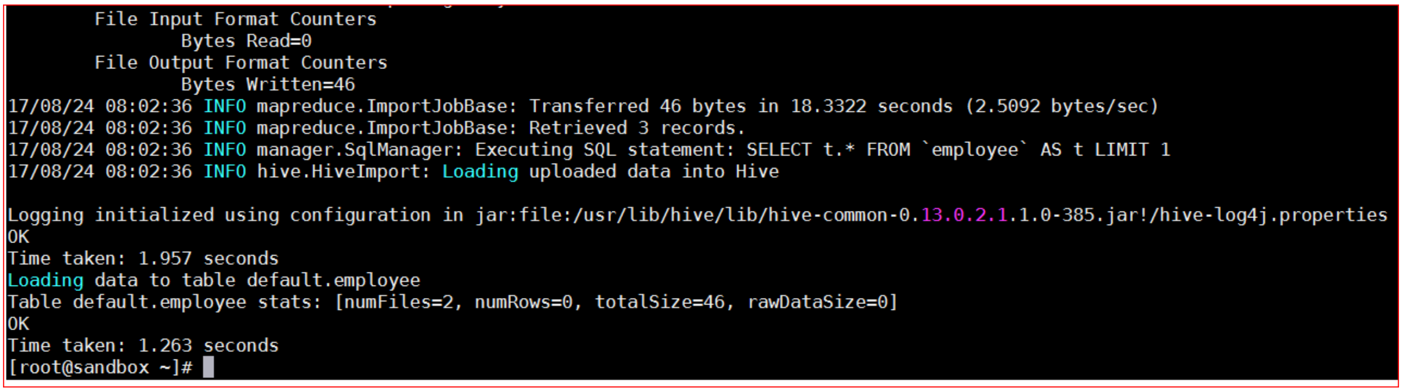
*--fields-terminated-by , *

*--hive-import *

-m 1



```
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'employee' \
> --target-dir '/sqoopout' \
> --columns id,name,salary \
> --fields-terminated-by , \
> --hive-import \
> -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.
17/08/24 08:02:10 INFO sqoop.Sqoop: Running Sqoop version: 1.4.4.2.1.1.0-385
Enter password:
17/08/24 08:02:12 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
17/08/24 08:02:12 INFO tool.CodeGenTool: Beginning code generation
17/08/24 08:02:14 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 08:02:14 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 08:02:14 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/46422fb871fd996895433824a0d9b2ec/employee.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```

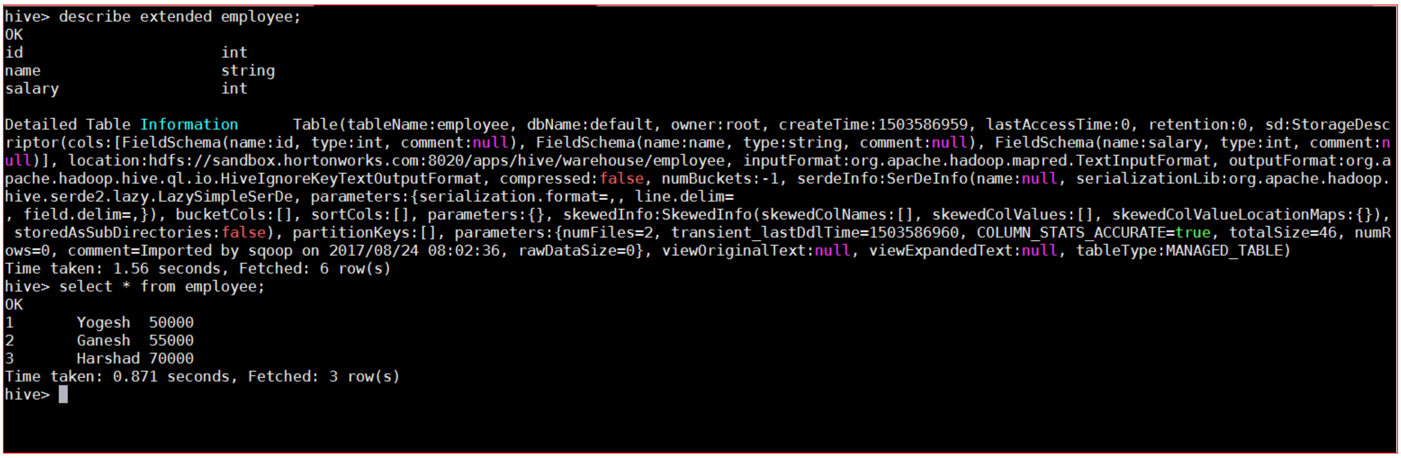


```
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=46
17/08/24 08:02:36 INFO mapreduce.ImportJobBase: Transferred 46 bytes in 18.3322 seconds (2.5092 bytes/sec)
17/08/24 08:02:36 INFO mapreduce.ImportJobBase: Retrieved 3 records.
17/08/24 08:02:36 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employee` AS t LIMIT 1
17/08/24 08:02:36 INFO hive.HiveImport: Loading uploaded data into Hive

Logging initialized using configuration in jar:file:/usr/lib/hive/lib/hive-common-0.13.0.2.1.1.0-385.jar!/hive-log4j.properties
OK
Time taken: 1.957 seconds
Loading data to table default.employee
Table default.employee stats: [numFiles=2, numRows=0, totalSize=46, rawDataSize=0]
OK
Time taken: 1.263 seconds
[root@sandbox ~]# █
```

Check table employee in hive:

*select * from employee;*

describe extended employee;



```
hive> describe extended employee;
OK
id                      int
name                    string
salary                  int

Detailed Table Information      Table(tableName:employee, dbName:default, owner:root, createTime:1503586959, lastAccessTime:0, retention:0, sd:StorageDesc
riptor(cols:[FieldSchema(name:id, type:int, comment:null), FieldSchema(name:name, type:string, comment:null), FieldSchema(name:salary, type:int, comment:n
ull)], location:hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/employee, inputFormat:org.apache.hadoop.mapred.TextInputFormat, outputFormat:org.a
pache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat, compressed:false, numBuckets:-1, serdeInfo:SerDeInfo(name:null, serializationLib:org.apache.hadoop.
hive.serde2.lazy.LazySimpleSerDe, parameters:{serialization.format=,, line.delim=
, field.delim=,}), bucketCols:[], sortCols:[], parameters:{}, skewedInfo:SkewedInfo(skewedColNames:[], skewedColValues:[], skewedColValueLocationMaps:{}),
 storedAsSubDirectories:false), partitionKeys:[], parameters:{numFiles=2, transient_lastDdlTime=1503586960, COLUMN_STATS_ACCURATE=true, totalSize=46, numR
ows=0, comment=Imported by sqoop on 2017/08/24 08:02:36, rawDataSize=0}, viewOriginalText:null, viewExpandedText:null, tableType:MANAGED_TABLE)
Time taken: 1.56 seconds, Fetched: 6 row(s)
hive> select * from employee;
OK
1       Yogesh  50000
2       Ganesh  55000
3       Harshad 70000
Time taken: 0.871 seconds, Fetched: 3 row(s)
hive>
```

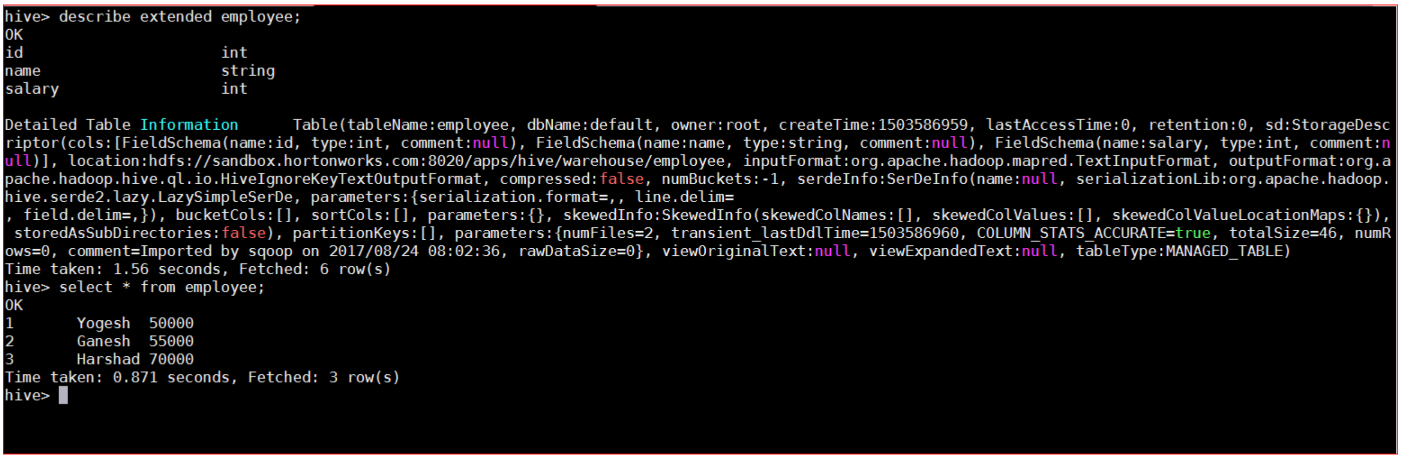
Data associated with selected columns have been imported successfully.

2. Export selected columns from hive to SQL

Check table employee in hive:

*select * from employee;*

describe extended employee;

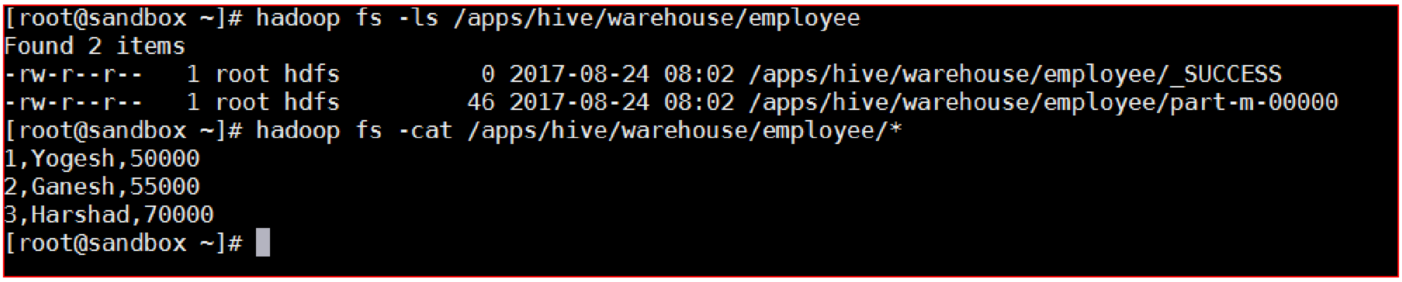


```
hive> describe extended employee;
OK
id                      int
name                    string
salary                  int

Detailed Table Information      Table(tableName:employee, dbName:default, owner:root, createTime:1503586959, lastAccessTime:0, retention:0, sd:StorageDesc
riptor(cols:[FieldSchema(name:id, type:int, comment:null), FieldSchema(name:name, type:string, comment:null), FieldSchema(name:salary, type:int, comment:n
ull)], location:hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/employee, inputFormat:org.apache.hadoop.mapred.TextInputFormat, outputFormat:org.a
pache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat, compressed:false, numBuckets:-1, serdeInfo:SerDeInfo(name:null, serializationLib:org.apache.hadoop.
hive.serde2.lazy.LazySimpleSerDe, parameters:{serialization.format=,, line.delim=
, field.delim=,}), bucketCols:[], sortCols:[], parameters:{}, skewedInfo:SkewedInfo(skewedColNames:[], skewedColValues:[], skewedColValueLocationMaps:{}),
 storedAsSubDirectories:false), partitionKeys:[], parameters:{numFiles=2, transient_lastDdlTime=1503586960, COLUMN_STATS_ACCURATE=true, totalSize=46, numR
ows=0, comment=Imported by sqoop on 2017/08/24 08:02:36, rawDataSize=0}, viewOriginalText:null, viewExpandedText:null, tableType:MANAGED_TABLE)
Time taken: 1.56 seconds, Fetched: 6 row(s)
hive> select * from employee;
OK
1       Yogesh  50000
2       Ganesh  55000
3       Harshad 70000
Time taken: 0.871 seconds, Fetched: 3 row(s)
hive>
```

Check file in HDFS:

# hadoop fs -ls /apps/hive/warehouse/employee

*# hadoop fs -cat /apps/hive/warehouse/employee/**



```
[root@sandbox ~]# hadoop fs -ls /apps/hive/warehouse/employee
Found 2 items
-rw-r--r--   1 root hdfs          0 2017-08-24 08:02 /apps/hive/warehouse/employee/_SUCCESS
-rw-r--r--   1 root hdfs         46 2017-08-24 08:02 /apps/hive/warehouse/employee/part-m-00000
[root@sandbox ~]# hadoop fs -cat /apps/hive/warehouse/employee/*
1,Yogesh,50000
2,Ganesh,55000
3,Harshad,70000
[root@sandbox ~]#
```

Create table employeeExport with column as id and name:

use db1;

show tables;

create table employeeExport

(

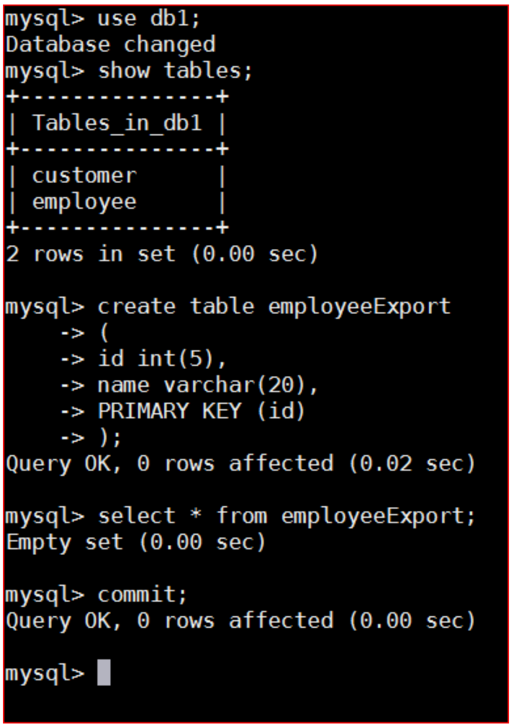
id int(5),

name varchar(20),

PRIMARY KEY (id)

);

*select * from employeeExport;*



```
mysql> use db1;
Database changed
mysql> show tables;
+---------------+
| Tables_in_db1 |
+---------------+
| customer      |
| employee      |
+---------------+
2 rows in set (0.00 sec)

mysql> create table employeeExport
    -> (
    -> id int(5),
    -> name varchar(20),
    -> PRIMARY KEY (id)
    -> );
Query OK, 0 rows affected (0.02 sec)

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

mysql> commit;
Query OK, 0 rows affected (0.00 sec)

mysql> 
```

Run below sqoop export command for getting selected columns in mysql.

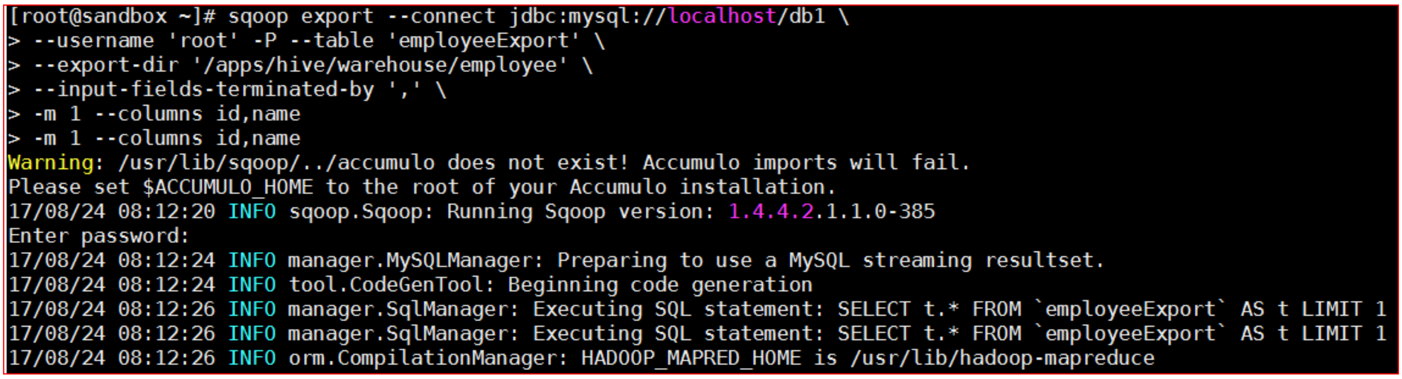
*sqoop export --connect jdbc:mysql://localhost/db1 *

*--username 'root' -P --table 'employeeExport' *

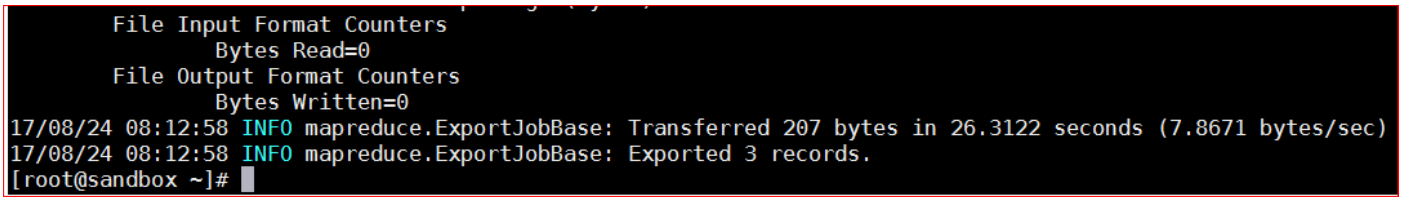
*--export-dir '/apps/hive/warehouse/employee' *

*--input-fields-terminated-by ',' *

-m 1 --columns id,name



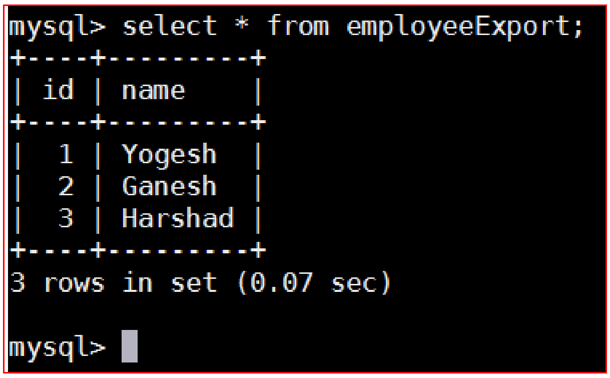
```
[root@sandbox ~]# sqoop export --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'employeeExport' \
> --export-dir '/apps/hive/warehouse/employee' \
> --input-fields-terminated-by ',' \
> -m 1 --columns id,name
> -m 1 --columns id,name
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
17/08/24 08:12:20 INFO sqoop.Sqoop: Running Sqoop version: 1.4.4.2.1.1.0-385
Enter password:
17/08/24 08:12:24 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
17/08/24 08:12:24 INFO tool.CodeGenTool: Beginning code generation
17/08/24 08:12:26 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employeeExport` AS t LIMIT 1
17/08/24 08:12:26 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `employeeExport` AS t LIMIT 1
17/08/24 08:12:26 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
```



```
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
17/08/24 08:12:58 INFO mapreduce.ExportJobBase: Transferred 207 bytes in 26.3122 seconds (7.8671 bytes/sec)
17/08/24 08:12:58 INFO mapreduce.ExportJobBase: Exported 3 records.
[root@sandbox ~]#
```

Check table employeeExport in SQL:

*select * from employeeExport;*



```
mysql> select * from employeeExport;
+----+---------+
| id | name    |
+----+---------+
|  1 | Yogesh  |
|  2 | Ganesh  |
|  3 | Harshad |
+----+---------+
3 rows in set (0.07 sec)

mysql>
```

Selected column data have exported successfully from hive to SQL.