Bigdata Assignment 4.4

Perform incremental load in Hive.Read from MySQL Table and load it in Hive table.Create hive table if it does not exist.If it exists, perform the incremental load.

Solution -

- Created a database 'acadgild' and inside it a table 'student' where 5 records where inserted
  create database acadgild;
  use acadgild;
  create table student (id int , name varchar(20) , city varchar(20));

```
mysql> create database acadgild;
Query OK, 1 row affected (0.03 sec)

mysql> use acadgild;
Database changed
mysql> create table student(id int ,name varchar(20),city varchar(20));
Query OK, 0 rows affected (0.03 sec)

mysql> insert into  student values
    -> (1,'ritesh','bbsr'),
    -> (2,'shyam','pune'),
    -> (3,'ram','agra'),
    -> (4,'hari','noida'),
    -> (5,'alok','delhi');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql>
```

- Checked the contensts of the table.
  select * from student;

```
mysql> select * from student;
+------+--------+-------+
| id   | name   | city  |
+------+--------+-------+
|    1 | ritesh | bbsr  |
|    2 | shyam  | pune  |
|    3 | ram    | agra  |
|    4 | hari   | noida |
|    5 | alok   | delhi |
+------+--------+-------+
5 rows in set (0.00 sec)
```

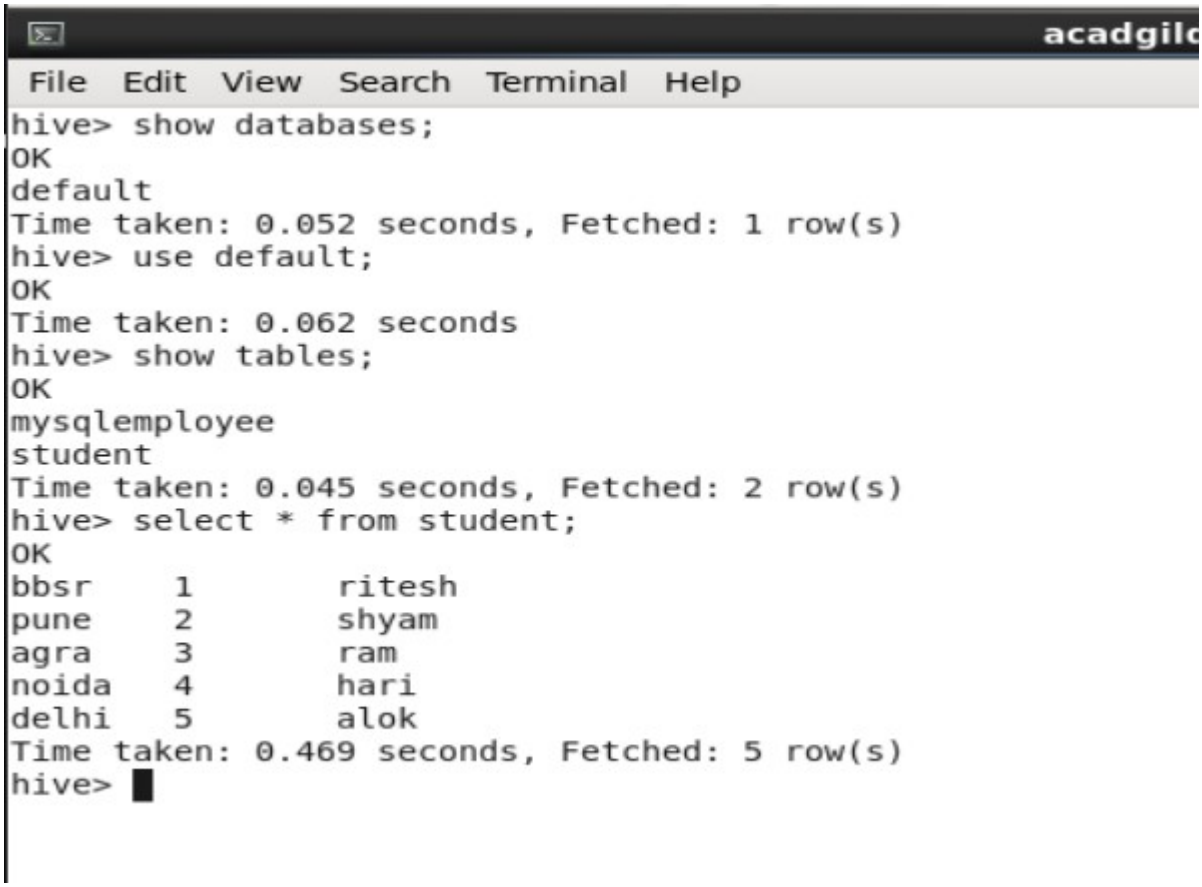
- In hive , we checked the default database , no tudent table exists.
  show databases;
  show tables;

```
hive> show databases;
OK
default
Time taken: 8.979 seconds, Fetched: 1 row(s)
hive> use default;
OK
Time taken: 0.05 seconds
hive> show tables;
OK
mysqlemployee
Time taken: 0.092 seconds, Fetched: 1 row(s)
hive>
```

- We transferred the data from mysql to hive using sqoop.
  sqoop import –connect jdbc:mysql://localhost/acadgild –username root –password Root@123 –table student -m 1 –target-dir /sqoopouthive –hive-import

```
acadgild@localhost:~
[acadgild@localhost ~]$ sqoop import --connect jdbc:mysql://localhost/acadgild --username root --password Root@123  --table student -m 1 --target-dir /sqoopouthive --hive-import;
Warning: /home/acadgild/install/sqoop/sqoop-1.4.6.bin__hadoop-2.0.4-alpha/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /home/acadgild/install/sqoop/sqoop-1.4.6.bin__hadoop-2.0.4-alpha/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
18/06/14 01:57:30 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6
18/06/14 01:57:30 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
18/06/14 01:57:30 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can override
18/06/14 01:57:30 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
18/06/14 01:57:31 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
18/06/14 01:57:31 INFO tool.CodeGenTool: Beginning code generation
Thu Jun 14 01:57:31 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
18/06/14 01:57:33 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
18/06/14 01:57:33 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `student` AS t LIMIT 1
18/06/14 01:57:33 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /home/acadgild/install/hadoop/hadoop-2.6.5
Note: /tmp/sqoop-acadgild/compile/82a7075c2a268e1c3d9f3f1bcc647f5b/student.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
18/06/14 01:57:40 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-acadgild/compile/82a7075c2a268e1c3d9f3f1bcc647f5b/student.jar
18/06/14 01:57:40 WARN manager.MySQLManager: It looks like you are importing from mysql.
18/06/14 01:57:40 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
18/06/14 01:57:40 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
18/06/14 01:57:40 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
18/06/14 01:57:40 INFO mapreduce.ImportJobBase: Beginning import of student
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/acadgild/install/hbase/hbase-1.2.6/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]
18/06/14 01:57:40 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

- Then , in hive we checked the content of student
  select * from student;



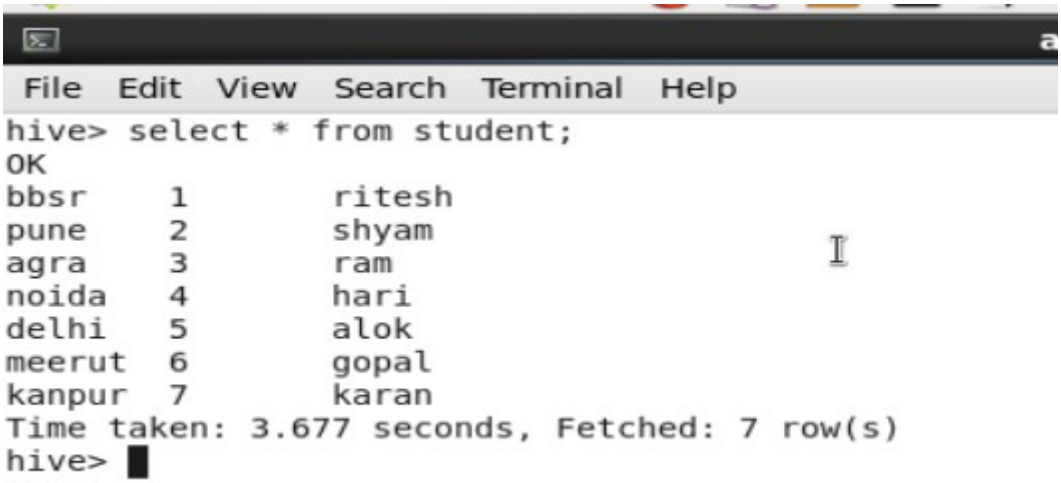
- Now , we added 2 records in the student table in mysql . Then checked its contents
  select * from student;

```
mysql> insert into  student values (6,'gopal','meerut'), (7,'karan','kanpur');
Query OK, 2 rows affected (0.08 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql> select * from student;
+------+--------+--------+
| id   | name   | city   |
+------+--------+--------+
|    1 | ritesh | bbsr   |
|    2 | shyam  | pune   |
|    3 | ram    | agra   |
|    4 | hari   | noida  |
|    5 | alok   | delhi  |
|    6 | gopal  | meerut |
|    7 | karan  | kanpur |
+------+--------+--------+
7 rows in set (0.00 sec)
```

- Then we performed the incremental load by using sqoop incremental
  sqoop import –connect jdbc:mysql//localhost/acadgild –username root –password Root@123 –table student –check-column id –incremental append –last-value 5 -m 1 –target-dir /sqooputhive;

- In hive , we checked the content of the table after incremental load **select * from student;**



The data is same as in the mysql after updating 2 records.