



Apache Sqoop

By Sumit Mittal



Apache Sqoop

Exercise 1



IMPORTANT

Copyright Infringement and Illegal Content Sharing Notice

All course content designs, video, audio, text, graphics, logos, images are Copyright© and are protected by India and international copyright laws. All rights reserved.

Permission to download the contents (wherever applicable) for the sole purpose of individual reading and preparing yourself to crack the interview only. Any other use of study materials – including reproduction, modification, distribution, republishing, transmission, display – without the prior written permission of Author is strictly prohibited.

Trendytech Insights legal team, along with thousands of our students, actively searches the Internet for copyright infringements. Violators subject to prosecution.

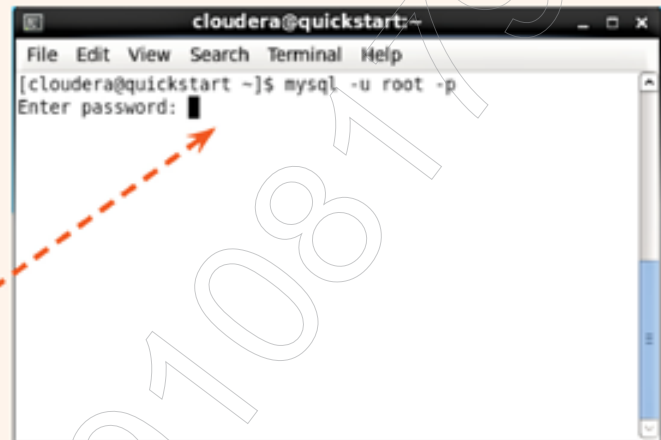
Sqoop Basics

To enter into MySQL:

```
mysql -u root -p
```

Note:

Enter password: cloudera

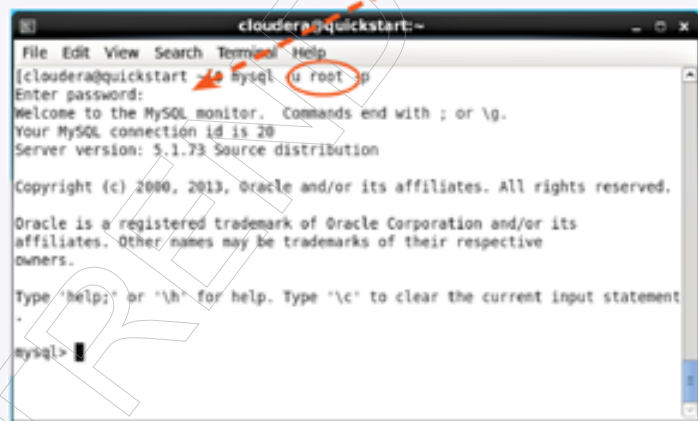


MySQL root user:

root user has access to all the databases.

```
mysql -u root -p
```

(Enter password: cloudera)

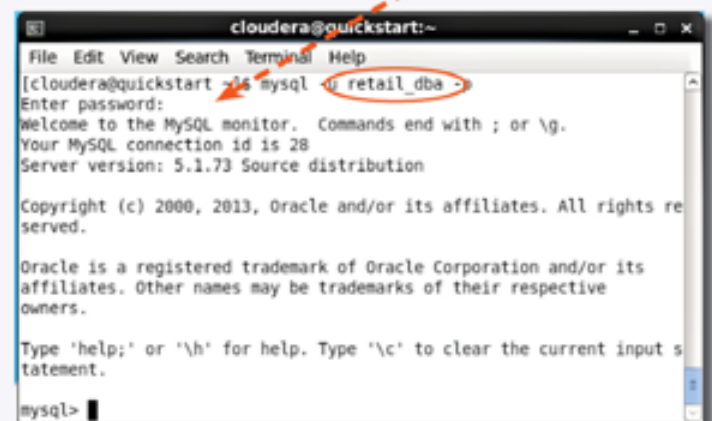


MySQL retail_dba user:

retail_dba user has access to limited databases.

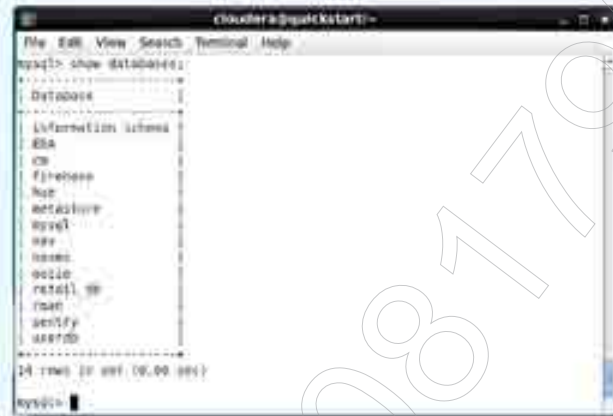
```
mysql -u retail_dba -p
```

(Enter password: cloudera)



To display databases in MySQL:

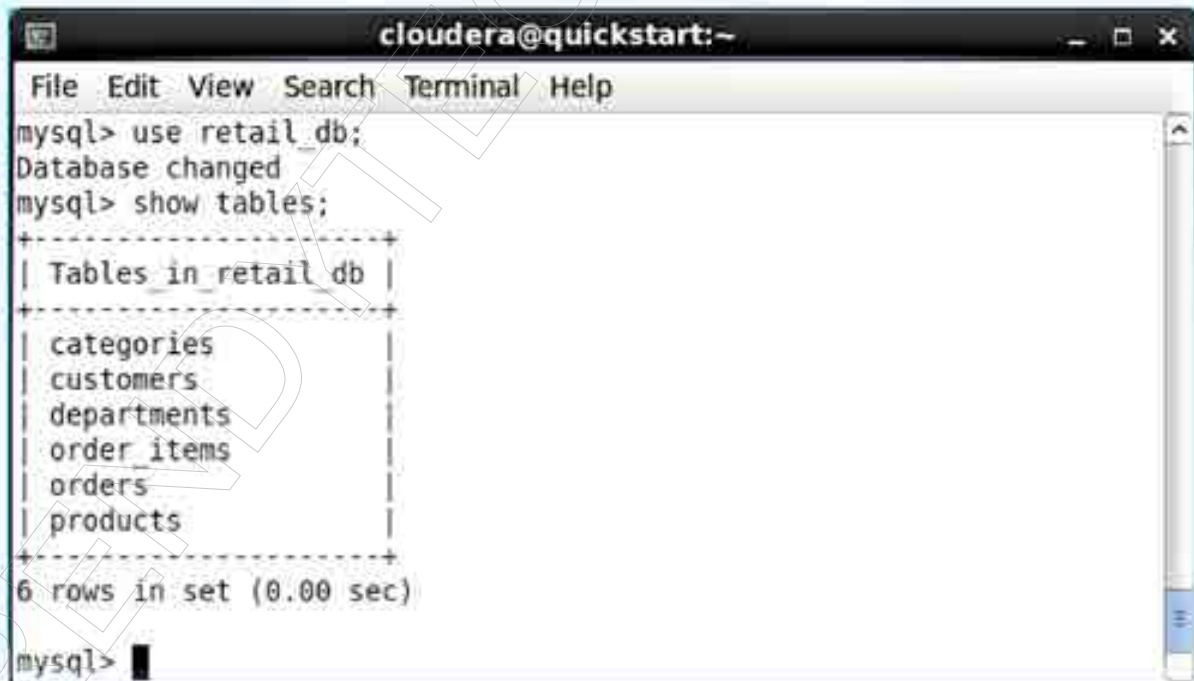
```
show databases;
```



Use databases and display tables:

```
use retail_db;
```

```
show tables;
```



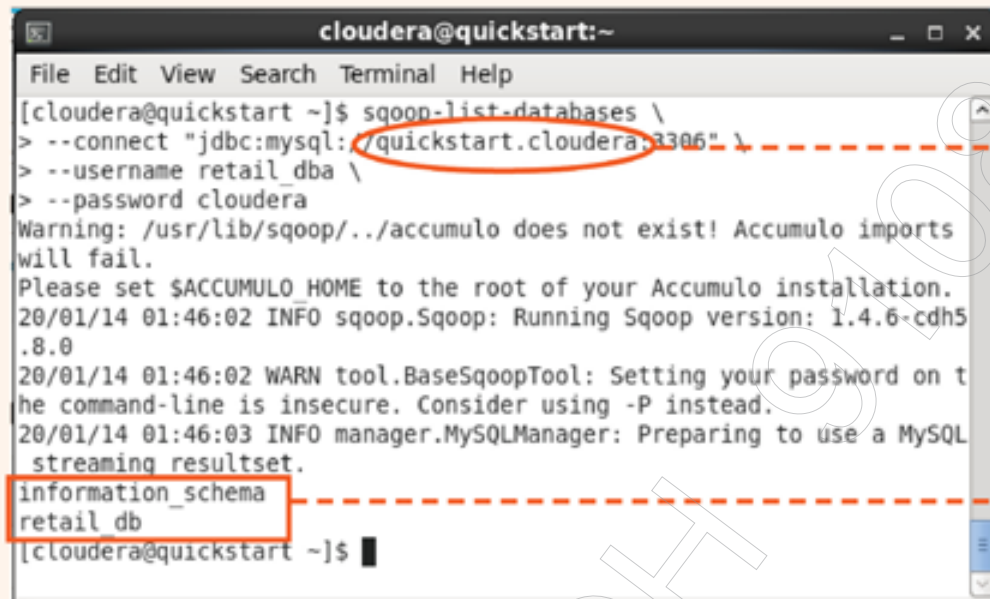
Accessing MySQL databases from Hadoop using Sqoop:

```
sqoop-list-databases \  
--connect "jdbc:mysql://quickstart.cloudera:3306" \  
--username retail_dba \  
--password cloudera
```

Space with backslash
(\) Indicates
continuation of line

Local host name

List of database

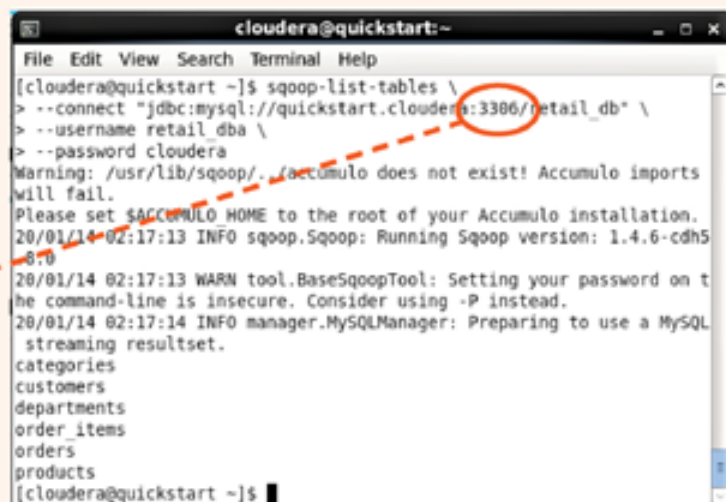


```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop-list-databases \  
> --connect "jdbc:mysql://quickstart.cloudera:3306" \  
> --username retail_dba \  
> --password cloudera  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports  
will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/14 01:46:02 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5  
.8.0  
20/01/14 01:46:02 WARN tool.BaseSqoopTool: Setting your password on t  
he command-line is insecure. Consider using -P instead.  
20/01/14 01:46:03 INFO manager.MySQLManager: Preparing to use a MySQL  
streaming resultset.  
information_schema  
retail_db  
[cloudera@quickstart ~]$
```

Accessing MySQL tables using the root user:

```
sqoop-list-tables \  
--connect "jdbc:mysql://quickstart.cloudera:3306/retail_db" \  
--username retail_dba \  
--password cloudera
```

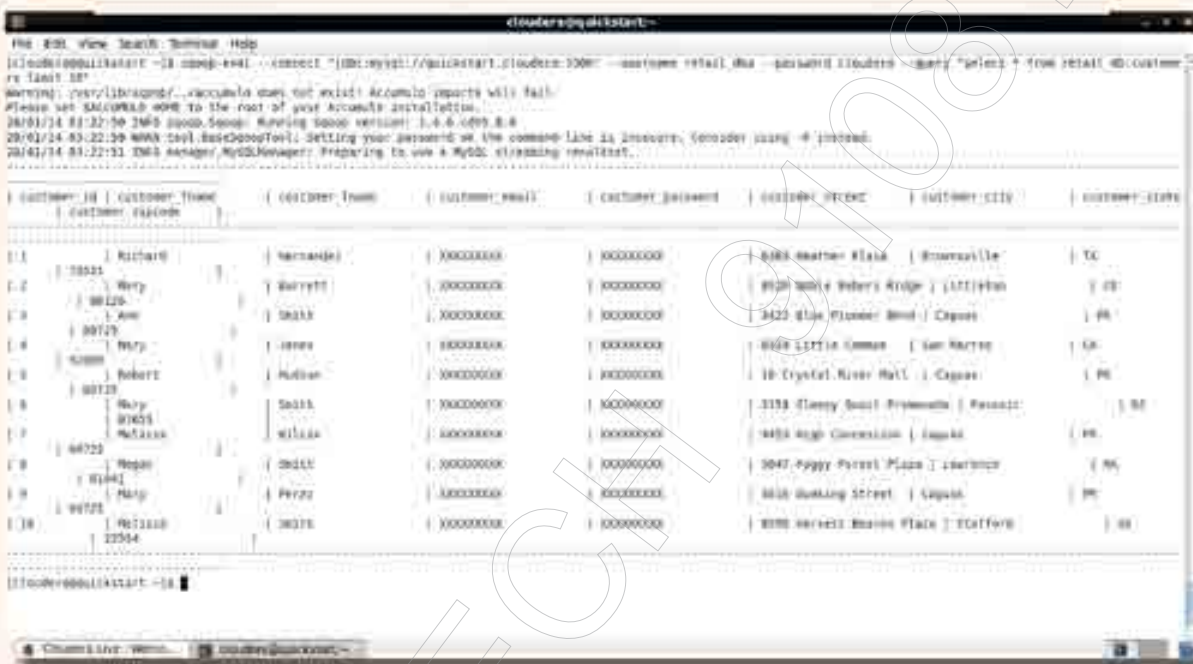
Local port no. where
MySQL runs



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop-list-tables \  
> --connect "jdbc:mysql://quickstart.cloudera:3306/retail_db" \  
> --username retail_dba \  
> --password cloudera  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports  
will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/14 02:17:13 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5  
.8.0  
20/01/14 02:17:13 WARN tool.BaseSqoopTool: Setting your password on t  
he command-line is insecure. Consider using -P instead.  
20/01/14 02:17:14 INFO manager.MySQLManager: Preparing to use a MySQL  
streaming resultset.  
categories  
customers  
departments  
order_items  
orders  
products  
[cloudera@quickstart ~]$
```

Displaying table data using sqoop-eval:

```
sqoop-eval \  
--connect "jdbc:mysql://quickstart.cloudera:3306" \  
--username retail_dba \  
--password cloudera \  
--query "select * from retail_db.customers limit 10"
```



CUSTOMER_ID	CUSTOMER_FIRST	CUSTOMER_LAST	CUSTOMER_EMAIL	CUSTOMER_PHONE	CUSTOMER_CITY	CUSTOMER_STATE
1	Michael	Bernardo	XXXXXXXXXX	XXXXXXXXXX	6485 Heather Plaza	TX
2	Wendy	Barryett	XXXXXXXXXX	XXXXXXXXXX	9928 Sandy Peters Ridge	TX
3	Sam	Smith	XXXXXXXXXX	XXXXXXXXXX	3422 Blue Plover Blvd	PA
4	Wendy	Jones	XXXXXXXXXX	XXXXXXXXXX	8058 Little Campus	CA
5	Robert	Mullan	XXXXXXXXXX	XXXXXXXXXX	18 Crystal River Mall	PA
6	Wendy	Smith	XXXXXXXXXX	XXXXXXXXXX	3158 Cleary Court	MA
7	Michael	Wilcox	XXXXXXXXXX	XXXXXXXXXX	4454 High Greenfield	PA
8	Wendy	Smith	XXXXXXXXXX	XXXXXXXXXX	3047 Pappy Peters Plaza	MA
9	Wendy	Perry	XXXXXXXXXX	XXXXXXXXXX	6616 Dunning Street	PA
10	Michael	Smith	XXXXXXXXXX	XXXXXXXXXX	8058 Sandy Peters Plaza	PA

Create and use a database in MySQL:

```
CREATE database trendytech;
```

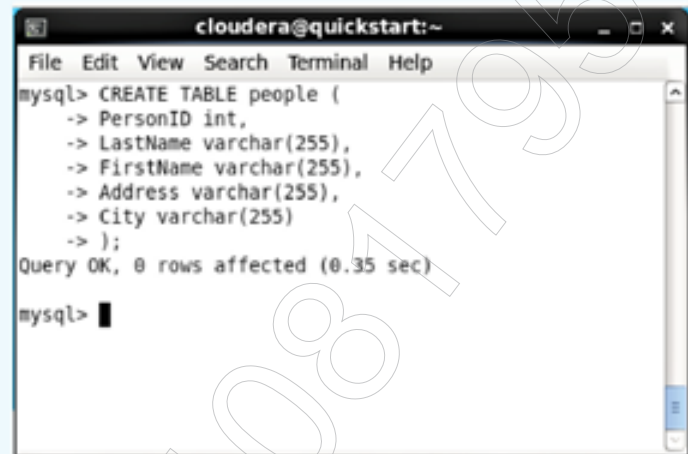
```
USE trendytech;
```



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
mysql> create database trendytech;  
Query OK, 1 row affected (0.18 sec)  
  
mysql> use trendytech;  
Database changed  
mysql> █
```

Create a table in MySQL:

```
CREATE TABLE people  
(  
  PersonID int,  
  LastName varchar(255),  
  FirstName varchar(255),  
  Address varchar(255),  
  City varchar(255)  
);
```

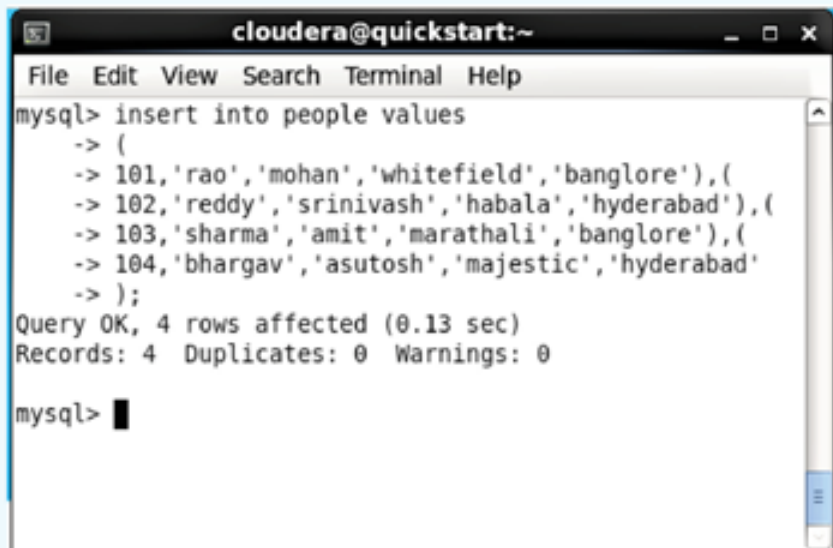


```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
mysql> CREATE TABLE people (  
-> PersonID int,  
-> LastName varchar(255),  
-> FirstName varchar(255),  
-> Address varchar(255),  
-> City varchar(255)  
-> );  
Query OK, 0 rows affected (0.35 sec)  
mysql>
```

Insert records into the people table:

```
insert into people values  
(  
  101,'rao','mohan','whitefield','bangalore'),(  
  102,'reddy','srinivash','habala','hyderabad'),(  
  103,'sharma','amit','marathali','bangalore'),(  
  104,'bhargav','asutosh','majestic','hyderabad'  
);
```

```
commit;
```

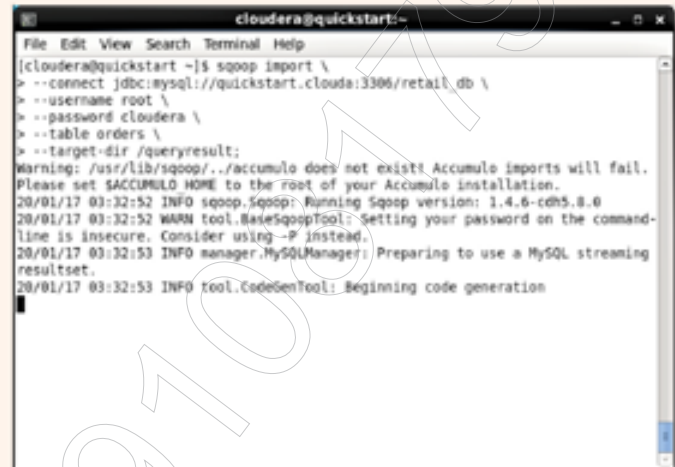


```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
mysql> insert into people values  
-> (  
-> 101,'rao','mohan','whitefield','bangalore'),(  
-> 102,'reddy','srinivash','habala','hyderabad'),(  
-> 103,'sharma','amit','marathali','bangalore'),(  
-> 104,'bhargav','asutosh','majestic','hyderabad'  
-> );  
Query OK, 4 rows affected (0.13 sec)  
Records: 4 Duplicates: 0 Warnings: 0  
mysql>
```


Import data from MySQL to Sqoop:

```
sqoop import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username root \  
--password cloudera \  
--table orders \  
--target-dir /queryresult
```

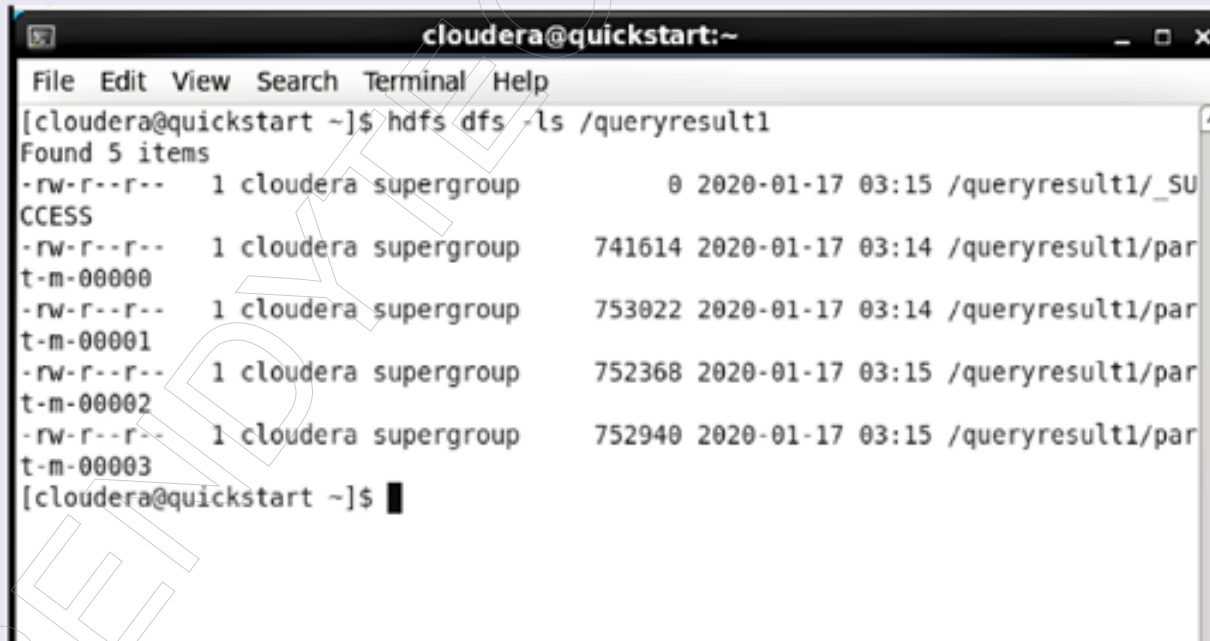
Note: If table don't have primary key than it will not import.



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop import \  
  --connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
  --username root \  
  --password cloudera \  
  --table orders \  
  --target-dir /queryresult;  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/17 03:32:52 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0  
20/01/17 03:32:52 WARN tool.BaseSqoopTool: Setting your password on the command-  
line is insecure. Consider using -P instead.  
20/01/17 03:32:53 INFO manager.MySQLManager: Preparing to use a MySQL streaming  
resultset.  
20/01/17 03:32:53 INFO tool.CodeGenTool: Beginning code generation
```

To display contents of queryresult directory in HDFS (use terminal):

```
hadoop fs -ls /queryresult
```



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ hdfs dfs -ls /queryresult1  
Found 5 items  
-rw-r--r-- 1 cloudera supergroup 0 2020-01-17 03:15 /queryresult1/_SU  
CESS  
-rw-r--r-- 1 cloudera supergroup 741614 2020-01-17 03:14 /queryresult1/par  
t-m-00000  
-rw-r--r-- 1 cloudera supergroup 753022 2020-01-17 03:14 /queryresult1/par  
t-m-00001  
-rw-r--r-- 1 cloudera supergroup 752368 2020-01-17 03:15 /queryresult1/par  
t-m-00002  
-rw-r--r-- 1 cloudera supergroup 752940 2020-01-17 03:15 /queryresult1/par  
t-m-00003  
[cloudera@quickstart ~]$
```

Note: By default the number of mappers are 4, so 4 output files are created.

Instructions ► Import the **people** table (which we have created earlier in MySQL) with same command as we did above.

To import people table from MySQL to HDFS:

```
sqoop import \
--connect jdbc:mysql://quickstart.cloudera:3306/trendytech \
--username root --password cloudera --table people \
--target-dir /peoplereult
```

```
cloudera@quickstart:~$ sqoop import --connect jdbc:mysql://quickstart.cloudera:3306/trendytech --username root --password cloudera --table people --target-dir /peoplereult
Warning: /usr/lib/sqoop/.../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
26/01/17 04:56:41 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0
26/01/17 04:56:41 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
26/01/17 04:56:42 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
26/01/17 04:56:42 INFO tool.CodeGenTool: Beginning code generation
26/01/17 04:56:44 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `people` AS t LIMIT 1
26/01/17 04:56:45 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `people` AS t LIMIT 1
26/01/17 04:56:45 INFO orm.CompilationManager: HADOOP MAPRED HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-cloudera/compile/49326ed3lac7fc2344ae1890299fe37b/people.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
26/01/17 04:56:50 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/49326ed3lac7fc2344ae1890299fe37b/people.jar
26/01/17 04:56:52 WARN manager.MySQLManager: It looks like you are importing from mysql.
26/01/17 04:56:52 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
26/01/17 04:56:52 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
26/01/17 04:56:52 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
26/01/17 04:56:54 ERROR tool.ImportTool: Error during import: No primary key could be found for table people. Please specify one with --split-by or perform a sequential import with '-m 1'.
cloudera@quickstart:~$
```

NOTE: it will throw error. Because **people** table doesn't have primary key.

Instructions ► Now, run the above command with mapper (**-m 1**):

To import people table from MySQL to HDFS with one Mapper:

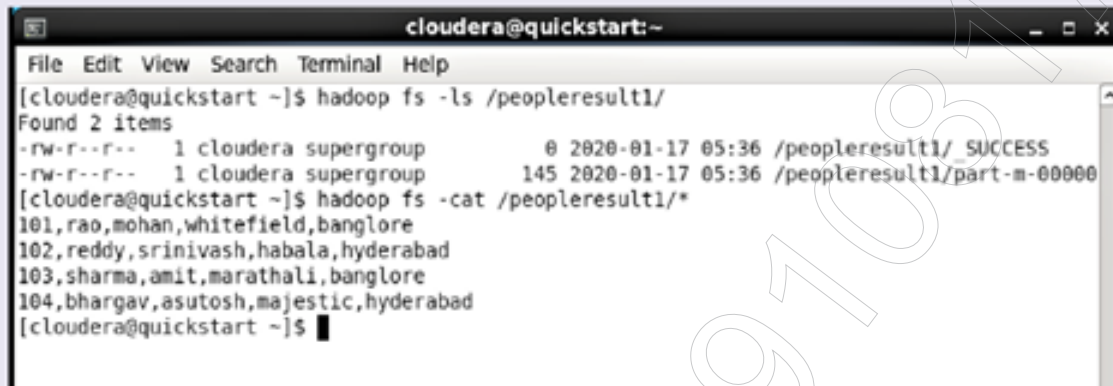
```
sqoop import \
--connect jdbc:mysql://quickstart.cloudera:3306/trendytech \
--username root \
--password cloudera \
--table people \
-m 1 \
--target-dir /peoplereult1
```

```
cloudera@quickstart:~$ sqoop import \
--connect jdbc:mysql://quickstart.cloudera:3306/trendytech \
--username root \
--password cloudera \
--table people \
-m 1 \
--target-dir /peoplereult1
Warning: /usr/lib/sqoop/.../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
26/01/17 05:37:00 INFO mapreduce.ImportJobBase: Transferred 145 bytes in 542.823
6 seconds (0.2673 bytes/sec)
26/01/17 05:37:00 INFO mapreduce.ImportJobBase: Retrieved 4 records.
cloudera@quickstart:~$
```

To display people table from HDFS:

```
hadoop fs -ls /peopleresult1
```

```
hadoop fs -cat /peopleresult1/*
```



The screenshot shows a terminal window titled 'cloudera@quickstart:~'. It displays the execution of two Hadoop commands. The first command, 'hadoop fs -ls /peopleresult1/', lists the contents of the directory, showing two items: a directory 'SUCCESS' and a file 'part-m-00000'. The second command, 'hadoop fs -cat /peopleresult1/*', concatenates the contents of the file 'part-m-00000', displaying a list of names and addresses.

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ hadoop fs -ls /peopleresult1/  
Found 2 items  
-rw-r--r-- 1 cloudera supergroup 0 2020-01-17 05:36 /peopleresult1/SUCCESS  
-rw-r--r-- 1 cloudera supergroup 145 2020-01-17 05:36 /peopleresult1/part-m-00000  
[cloudera@quickstart ~]$ hadoop fs -cat /peopleresult1/*  
101,rao,mohan,whitefield,banglore  
102,reddy,srinivash,habala,hyderabad  
103,sharma,amit,marathali,banglore  
104,bhargav,asutosh,majestic,hyderabad  
[cloudera@quickstart ~]$
```

Note: You will find one mapper file only (part-m-00000).

To import all tables from “MySQL” database:

```
sqoop-import-all-tables \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username retail_dba \  
--password cloudera \  
--as-sequencefile \ ←----- File Format  
-m 4 \  
--warehouse-dir /user/cloudera/sqoopdir
```

Note: Here no of mappers are 4 that means we will get 4 files

We can also mention **file format** while importing data as mentioned above.

Sqoop supports **4 types** of file formats:

- Text file format
- Sequence file format
- Avro file format
- Parquet file format

Note: If you do not mention any file format, by default it will be text file format.

By default Sqoop provides 4 mappers - so we can skip the above **-m 4** command and still get the same result.

Difference between Sqoop **target directory** & **warehouse directory**.

The difference is that:

-target-dir is a full directory path and the data files will be created directly inside the specified folder.

-warehouse-dir is used to specify a base directory within hdfs where SGOOP will create a sub folder inside with the name of the source table, and import the data files into that folder.

Directory structure for **retail_db** will be:

/user/cloudera/sqoopdir/employee
/user/cloudera/sqoopdir/customer
/user/cloudera/sqoopdir/table3
/user/cloudera/sqoopdir/tablw4

Now try to run following code to import the *orders* table with **--warehouse-dir** path:

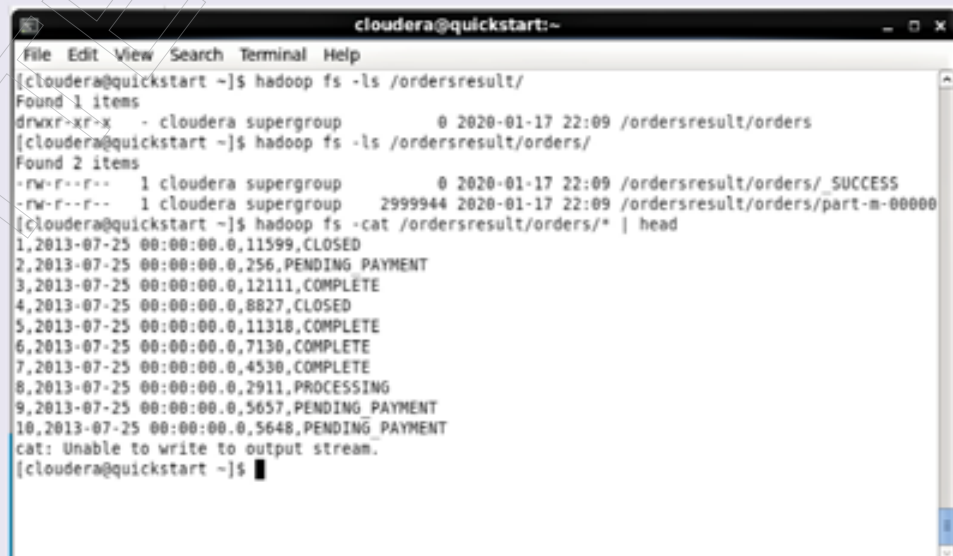
```
sqoop import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username root \  
--password cloudera \  
--table orders \  
--warehouse-dir /ordersresult
```

To check the file structure in HDFS:

```
hadoop fs -ls /ordersresult/
```

/user/cloudera/warehouse/ordersresult/orders

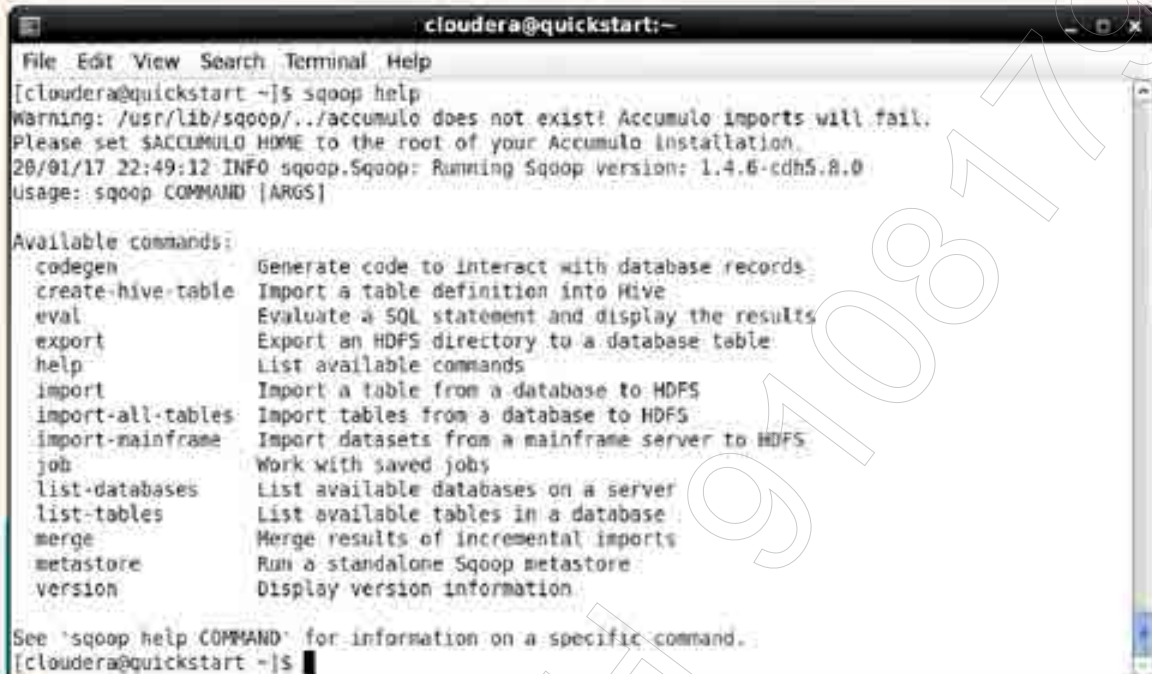
part-m-00000_0



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ hadoop fs -ls /ordersresult/  
Found 1 items  
drwxr-xr-x - cloudera supergroup 0 2020-01-17 22:09 /ordersresult/orders  
[cloudera@quickstart ~]$ hadoop fs -ls /ordersresult/orders/  
Found 2 items  
-rw-r--r-- 1 cloudera supergroup 0 2020-01-17 22:09 /ordersresult/orders/_SUCCESS  
-rw-r--r-- 1 cloudera supergroup 2999944 2020-01-17 22:09 /ordersresult/orders/part-m-00000  
[cloudera@quickstart ~]$ hadoop fs -cat /ordersresult/orders/* | head  
1,2013-07-25 00:00:00.0,11599,CLOSED  
2,2013-07-25 00:00:00.0,256,PENDING PAYMENT  
3,2013-07-25 00:00:00.0,12111,COMPLETE  
4,2013-07-25 00:00:00.0,8827,CLOSED  
5,2013-07-25 00:00:00.0,11318,COMPLETE  
6,2013-07-25 00:00:00.0,7130,COMPLETE  
7,2013-07-25 00:00:00.0,4530,COMPLETE  
8,2013-07-25 00:00:00.0,2911,PROCESSING  
9,2013-07-25 00:00:00.0,5657,PENDING PAYMENT  
10,2013-07-25 00:00:00.0,5648,PENDING PAYMENT  
cat: Unable to write to output stream.  
[cloudera@quickstart ~]$
```

To display a list of all available tools:

`sqoop help`



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop help  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/17 22:49:12 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0  
Usage: sqoop COMMAND [ARGS]  
  
Available commands:  
codegen          Generate code to interact with database records  
create-hive-table Import a table definition into Hive  
eval             Evaluate a SQL statement and display the results  
export           Export an HDFS directory to a database table  
help            List available commands  
import           Import a table from a database to HDFS  
import-all-tables Import tables from a database to HDFS  
import-mainframe Import datasets from a mainframe server to HDFS  
job             Work with saved jobs  
list-databases   List available databases on a server  
list-tables      List available tables in a database  
merge           Merge results of incremental imports  
metastore        Run a standalone Sqoop metastore  
version         Display version information  
  
See 'sqoop help COMMAND' for information on a specific command.  
[cloudera@quickstart ~]$
```

To know sqoop version:

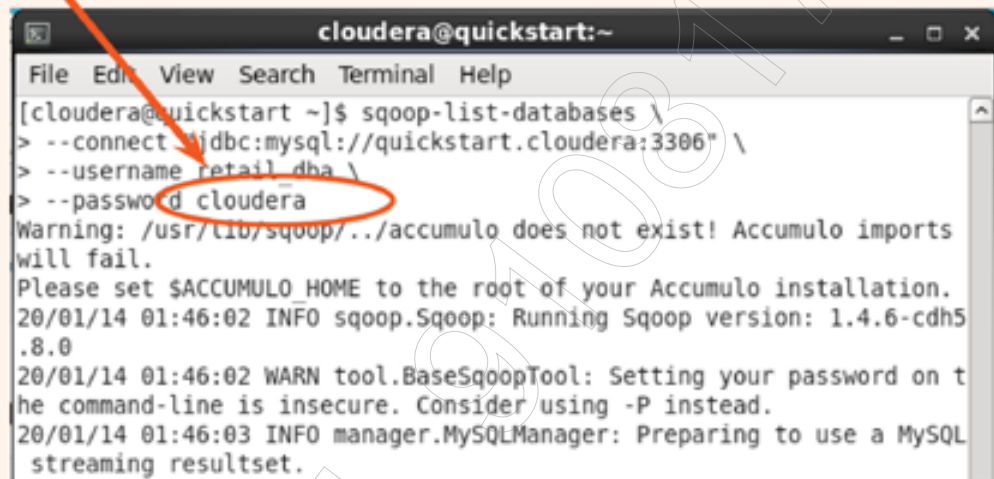
`sqoop version`



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop version  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/17 23:01:50 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0  
Sqoop 1.4.6-cdh5.8.0  
git commit id  
Compiled by jenkins on Thu Jun 16 12:25:21 PDT 2016  
[cloudera@quickstart ~]$
```


The argument **--password** takes authentication password in plain text.

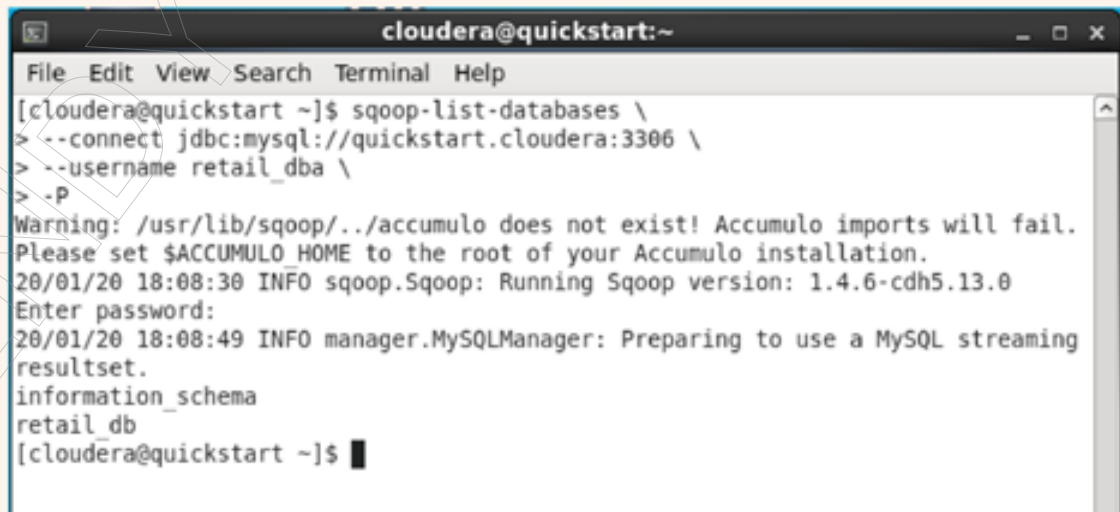
```
sqoop-list-databases \  
--connect jdbc:mysql://quickstart.cloudera:3306 \  
--username retail_dba \  
--password cloudera
```



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop-list-databases \  
> --connect jdbc:mysql://quickstart.cloudera:3306 \  
> --username retail_dba \  
> --password cloudera  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports  
will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/14 01:46:02 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5  
.8.0  
20/01/14 01:46:02 WARN tool.BaseSqoopTool: Setting your password on t  
he command-line is insecure. Consider using -P instead.  
20/01/14 01:46:03 INFO manager.MySQLManager: Preparing to use a MySQL  
streaming resultset.
```

While the argument **-P** read password from console.

```
sqoop-list-databases \  
--connect jdbc:mysql://quickstart.cloudera:3306 \  
--username retail_dba \  
-P
```



```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ sqoop-list-databases \  
> --connect jdbc:mysql://quickstart.cloudera:3306 \  
> --username retail_dba \  
> -P  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
20/01/20 18:08:30 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0  
Enter password:  
20/01/20 18:08:49 INFO manager.MySQLManager: Preparing to use a MySQL streaming  
resultset.  
information_schema  
retail_db  
[cloudera@quickstart ~]$
```


The argument **--query** can be replaced with **-e**.

```
sqoop-eval \  
--connect jdbc:mysql://quickstart.cloudera:3306 \  
--username retail_dba \  
--password cloudera \  
--query "select * from retail_db.customers limit 10"
```

OR

```
sqoop-eval \  
--connect "jdbc:mysql://quickstart.cloudera:3306" \  
--username retail_dba \  
--password cloudera \  
-e "select * from retail_db.customers limit 10"
```

Similarly **-m** and **--num-mappers** are same.

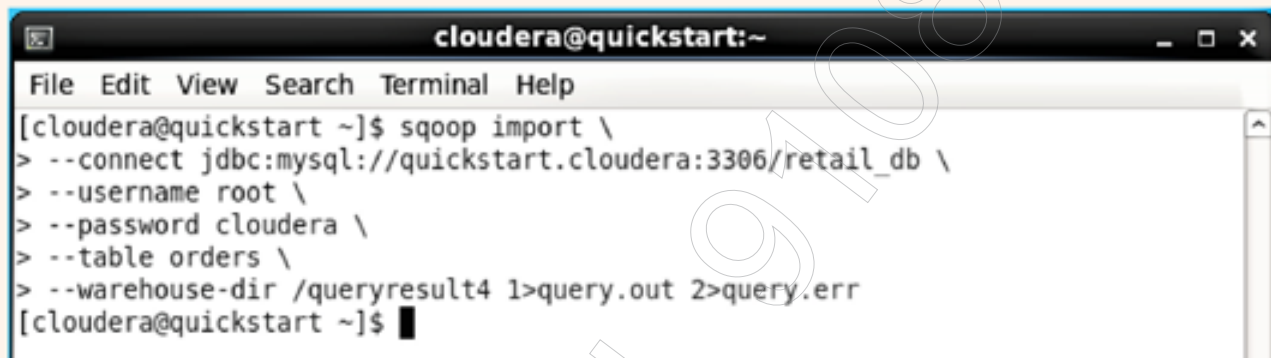
```
sqoop import \  
--connect jdbc:mysql://quickstart.cloudera:3306/trendytech \  
--username root \  
--password cloudera \  
--table people -m 1 \  
--target-dir /peoplereult1
```

OR

```
sqoop import \  
--connect jdbc:mysql://quickstart.cloudera:3306/trendytech \  
--username root \  
--password cloudera \  
--table people --num-mappers 1 \  
--target-dir /peoplereult1
```

Redirecting logs:

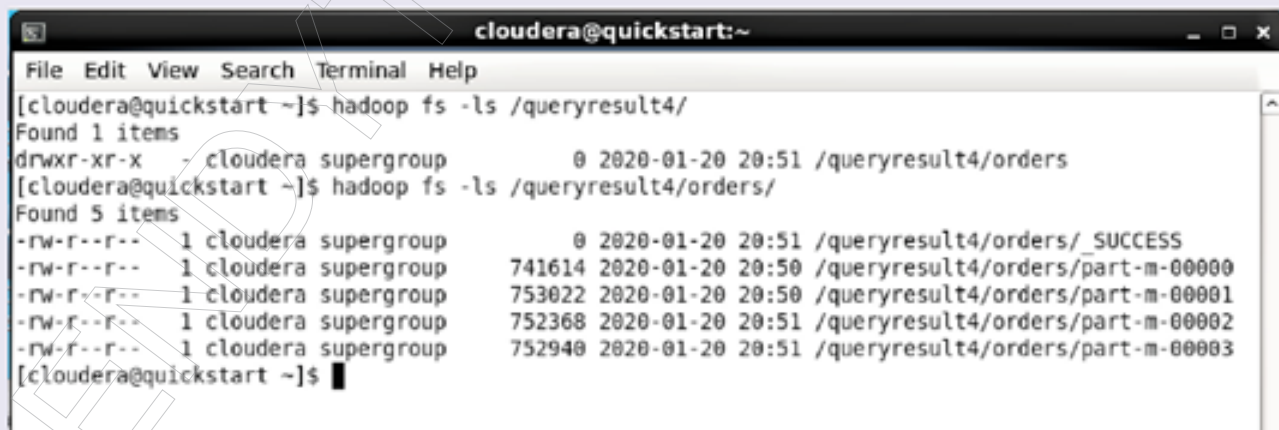
```
sqoop import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username root \  
--password cloudera \  
--table orders \  
--warehouse-dir /queryresult4 1>query.out 2>query.err
```



A terminal window titled 'cloudera@quickstart:~' showing the execution of the sqoop import command. The command is entered line by line, and the prompt returns after each line. The final command is executed, and the prompt returns. The command is: `sqoop import --connect jdbc:mysql://quickstart.cloudera:3306/retail_db --username root --password cloudera --table orders --warehouse-dir /queryresult4 1>query.out 2>query.err`

To check the content of the queryresult4:

```
hadoop fs -ls /queryresult4/orders/
```



A terminal window titled 'cloudera@quickstart:~' showing the execution of the hadoop fs -ls command. The command is entered, and the prompt returns. The command is: `hadoop fs -ls /queryresult4/orders/`

```
Found 1 items  
drwxr-xr-x 1 cloudera supergroup 0 2020-01-20 20:51 /queryresult4/orders  
[cloudera@quickstart ~]$ hadoop fs -ls /queryresult4/orders/  
Found 5 items  
-rw-r--r-- 1 cloudera supergroup 0 2020-01-20 20:51 /queryresult4/orders/_SUCCESS  
-rw-r--r-- 1 cloudera supergroup 741614 2020-01-20 20:50 /queryresult4/orders/part-m-00000  
-rw-r--r-- 1 cloudera supergroup 753022 2020-01-20 20:50 /queryresult4/orders/part-m-00001  
-rw-r--r-- 1 cloudera supergroup 752368 2020-01-20 20:51 /queryresult4/orders/part-m-00002  
-rw-r--r-- 1 cloudera supergroup 752940 2020-01-20 20:51 /queryresult4/orders/part-m-00003  
[cloudera@quickstart ~]$
```

To check the contents of log files:

```
cat query.out
```

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ cat query.out  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
[cloudera@quickstart ~]$
```

```
cat query.err
```

```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ cat query.err  
20/01/20 20:49:05 INFO sqoop.Sqoop: Running Sqoop version: 1.4.0-cdh5.13.0  
20/01/20 20:49:05 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.  
20/01/20 20:49:05 INFO mapreduce.Job: Running job: job_2001200000000000000  
20/01/20 20:49:05 INFO mapreduce.Job: MapReduce Framework  
20/01/20 20:49:05 INFO mapreduce.Job:   Map input records=88883  
20/01/20 20:49:05 INFO mapreduce.Job:   Map output records=88883  
20/01/20 20:49:05 INFO mapreduce.Job:   Input split bytes=4096  
20/01/20 20:49:05 INFO mapreduce.Job:   Spilled Records=0  
20/01/20 20:49:05 INFO mapreduce.Job:   Failed Shuffles=0  
20/01/20 20:49:05 INFO mapreduce.Job:   Merged Map outputs=0  
20/01/20 20:49:05 INFO mapreduce.Job:   GC time elapsed (ms)=694  
20/01/20 20:49:05 INFO mapreduce.Job:   CPU time spent (ms)=24738  
20/01/20 20:49:05 INFO mapreduce.Job:   Physical memory (bytes) snapshot=620457472  
20/01/20 20:49:05 INFO mapreduce.Job:   Virtual memory (bytes) snapshot=6300893184  
20/01/20 20:49:05 INFO mapreduce.Job:   Total committed heap usage (bytes)=392091712  
20/01/20 20:49:05 INFO mapreduce.Job:   File Input Format Counters  
20/01/20 20:49:05 INFO mapreduce.Job:     Bytes Read=0  
20/01/20 20:49:05 INFO mapreduce.Job:   File Output Format Counters  
20/01/20 20:49:05 INFO mapreduce.Job:     Bytes Written=2999944  
20/01/20 20:51:00 INFO mapreduce.ImportJobBase: Transferred 2.861 MB in 102.9298 seconds (28.4624 KB/sec)  
20/01/20 20:51:00 INFO mapreduce.ImportJobBase: Retrieved 88883 records.  
[cloudera@quickstart ~]$
```


Sqoop import execution flow

How Mappers divide their work when a query fired:

- Selects 1 record and by using that it gets the metadata and builds the java file

```

cloudera@quickstart:~
File Edit View Search Terminal Help
[cloudera@quickstart ~]$ sqoop import \
> --connect jdbc:mysql://quickstart.cloudera:3306/retail_db \
> --username root \
> --password cloudera \
> --table orders \
> --warehouse-dir /queryresult5
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
20/01/20 23:08:23 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0
20/01/20 23:08:23 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
20/01/20 23:08:23 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
20/01/20 23:08:23 INFO tool.CodeGenTool: Beginning code generation
20/01/20 23:08:25 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `orders` AS t LIMIT 1
20/01/20 23:08:25 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `orders` AS t LIMIT 1
20/01/20 23:08:25 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce

```

- Using above java file it builds the jar file

```

20/01/20 23:08:25 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `orders` AS t LIMIT 1
20/01/20 23:08:25 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `orders` AS t LIMIT 1
20/01/20 23:08:25 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-cloudera/compile/8d692f80ec566e5b217ab5df20dbec7c/orders.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
20/01/20 23:08:29 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/8d692f80ec566e5b217ab5df20dbec7c/orders.jar
20/01/20 23:08:29 WARN manager.MySQLManager: It looks like you are importing from mysql.

```

- **BoundingValsQuery** based on min and max on primary key

```

20/01/20 23:08:33 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
20/01/20 23:08:39 INFO db.DBInputFormat: Using read committed transaction isolation
20/01/20 23:08:39 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN(`order_id`), MAX(`order_id`) FROM `orders`
20/01/20 23:08:39 INFO db.IntegersSplitter: Split size: 17220; Num splits: 4 from: 1 to: 68883
20/01/20 23:08:40 INFO mapreduce.JobSubmitter: number of splits:4
20/01/20 23:08:46 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1579528165914_0010
20/01/20 23:08:49 INFO impl.YarnClientImpl: Submitted application application_1579528165914_0010

```

- Calculates $(\max - \min)/4$ and it gets the split size.

```

20/01/20 23:08:33 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
20/01/20 23:08:39 INFO db.DBInputFormat: Using read committed transaction isolation
20/01/20 23:08:39 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN(`order_id`), MAX(`order_id`) FROM `orders`
20/01/20 23:08:39 INFO db.IntegersSplitter: Split size: 17220; Num splits: 4 from: 1 to: 68883
20/01/20 23:08:40 INFO mapreduce.JobSubmitter: number of splits:4
20/01/20 23:08:46 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1579528165914_0010
20/01/20 23:08:49 INFO impl.YarnClientImpl: Submitted application application_1579528165914_0010
20/01/20 23:08:49 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1579528165914_0010/
20/01/20 23:08:49 INFO mapreduce.Job: Running job: job_1579528165914_0010

```


File formats:

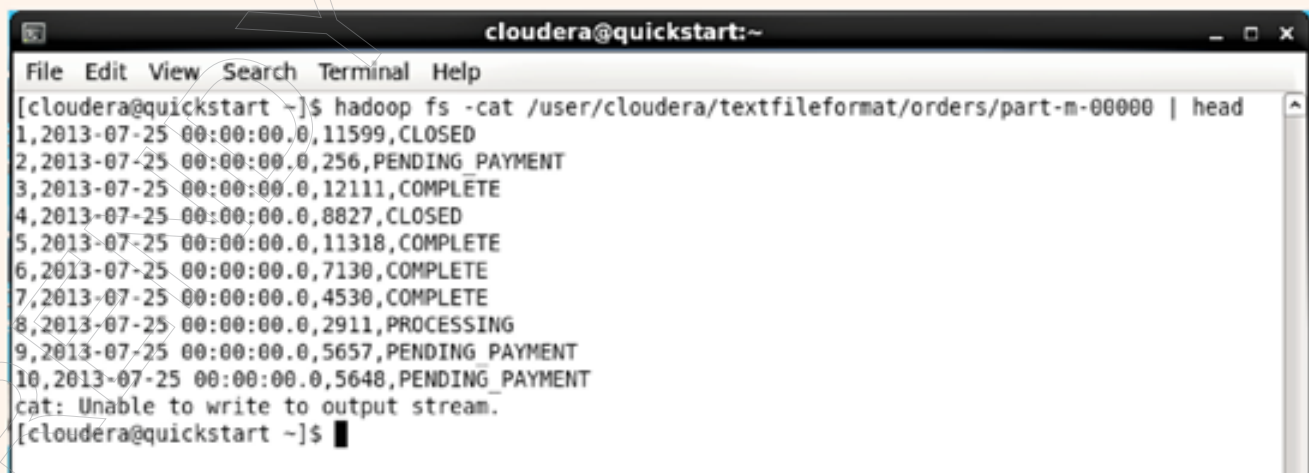
Sqoop import supports following file formats:

1. Text file format - command argument **--as-textfile**
2. Sequence file format - command argument **--as-sequencefile**
3. Avro file format - command argument **--as-avrodatafile**
4. Parquet file format - command argument **--as-parquetfile**

Note: If you are not mentioning any file format, by default sqoop uses **--as-textfile**

Text file format:

```
sqoop-import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username retail_dba \  
--password cloudera \  
--table orders \  
--as-textfile \  
-m 4 \  
--warehouse-dir /user/cloudera/textfileformat
```



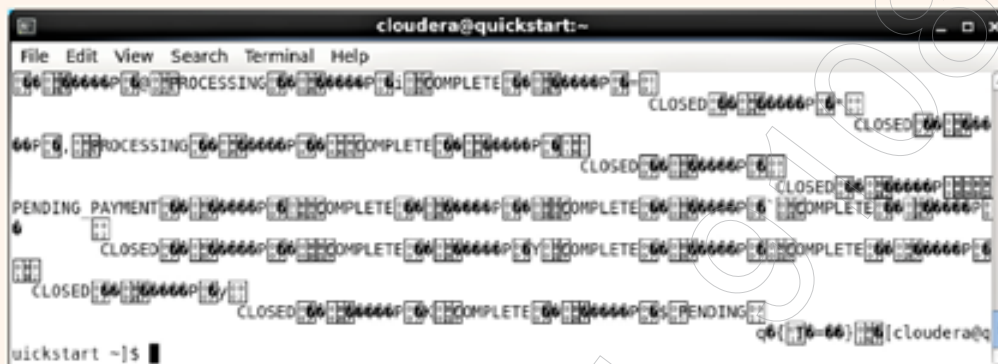
```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ hadoop fs -cat /user/cloudera/textfileformat/orders/part-m-00000 | head  
1,2013-07-25 00:00:00.0,11599,CLOSED  
2,2013-07-25 00:00:00.0,256,PENDING PAYMENT  
3,2013-07-25 00:00:00.0,12111,COMPLETE  
4,2013-07-25 00:00:00.0,8827,CLOSED  
5,2013-07-25 00:00:00.0,11318,COMPLETE  
6,2013-07-25 00:00:00.0,7130,COMPLETE  
7,2013-07-25 00:00:00.0,4530,COMPLETE  
8,2013-07-25 00:00:00.0,2911,PROCESSING  
9,2013-07-25 00:00:00.0,5657,PENDING PAYMENT  
10,2013-07-25 00:00:00.0,5648,PENDING PAYMENT  
cat: Unable to write to output stream.  
[cloudera@quickstart ~]$
```

```
File Edit View Search Terminal Help
[cloudera@quickstart ~]$ hadoop fs -cat /user/cloudera/sequencefileformat/orders/part-m-00000 | head ^
SEQ org.apache.hadoop.io.LongWritable orders(0:MFW)#6 00 000 000 0 CLOSED 000 000 PENDING_PAYMENT
000 000/COMPLETE 000 000 "CLOSED 000 000, COMPLETE 000 000 COMPLETE 000 000 COMPLET 000 000
PROCESSING0 000 000 PENDING_PAYMENT0
000 000 PENDING_PAYMENT/
000 000 PAYMENT_REVIEW"
000 000 CLOSED0
000 000 rPENDING PAYMENT+
PROCESSING000 000
COMPLETE 000 000 PENDING PAYMENT 000 000
COMPLETE 000 000 CLOSED 000 000 PENDING PAYMENT 000 000
PROCESSING 000 000
SPENDING 000 000 COMPLETE 000 000 PENDING PAYMENT 000 000, 000 000 CLOSED 000 000 CLOSED 000 000 COMPLETE
000 000
SPENDING PAYMENT 000 000 COMPLETE 000 000
PROCESSING 000 000 7PENDING PAYMENT 000 000 PAYMENT_REVIEW 000 000 COMPLETE ! 000 000 PENDING_PAYMENT!
"000 000
cat: Unable to write to output stream.
[cloudera@quickstart ~]$
```

Note: SequenceFiles are a binary format that store individual records in custom record-specific data types. These data types are manifested as Java classes.

Avro file format:

```
sqoop-import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username retail_dba --password cloudera \  
--table orders \  
--as-avrodatafile -m 4 \  
--warehouse-dir /user/cloudera/avrofileformat
```



Parquet file format:

```
sqoop-import \  
--connect jdbc:mysql://quickstart.cloudera:3306/retail_db \  
--username retail_dba --password cloudera \  
--table orders \  
--as-parquetfile -m 4 \  
--warehouse-dir /user/cloudera/parquetfileformat
```





5 Star Google Rated Big Data Course

LEARN FROM THE EXPERT



9108179578

Call for more details

Follow US

Trainer Mr. Sumit Mittal

LinkedIn <https://www.linkedin.com/in/bigdatabysumit/>

Website <https://trendytech.in/courses/big-data-online-training/>

Phone 9108179578

Email trendytech.sumit@gmail.com

Youtube TrendyTech

Twitter @BigdataBySumit

Instagram bigdatabysumit

Facebook <https://www.facebook.com/trendytech.in/>

