

This chapter describes how to import data from MySQL database to Hadoop HDFS. The „Import tool“ imports individual tables from RDBMS to HDFS. Each row in a table is treated as a record in HDFS. All records are stored as text data in the text files or as binary data in Avro and Sequence files.

Syntax

The following syntax is used to import data into HDFS.

```
$ sqoop import (generic-args) (import-args)
$ sqoop-import (generic-args) (import-args)
```

Example

Let us take an example of three tables named as **emp**, **emp_add**, and **emp_contact**, which are in a database called **userdb** in a MySQL database server.

The three tables and their data are as follows.

emp:

id	name	deg	salary	dept
1201	gopal	manager	50,000	TP
1202	manisha	Proof reader	50,000	TP
1203	khalil	php dev	30,000	AC
1204	prasanth	php dev	30,000	AC
1204	kranthi	admin	20,000	TP

emp_add:

id	hno	street	city
1201	288A	vgiri	jublee
1202	108I	aoc	sec-bad
1203	144Z	pgutta	hyd
1204	78B	old city	sec-bad
1205	720X	hitec	sec-bad

id	phno	email
1201	2356742	gopal@tp.com
1202	1661663	manisha@tp.com
1203	8887776	khalil@ac.com

1204 9988774 prasanth@ac.com
1205 1231231 kranthi@tp.com

Importing a Table

Sqoop tool „import“ is used to import table data from the table to the Hadoop file system as a text file or a binary file.

The following command is used to import the **emp** table from MySQL database server to HDFS.

```
$ sqoop import \  
--connect jdbc:mysql://localhost/userdb \  
--username root \  
--table emp --m 1
```

If it is executed successfully, then you get the following output.

```
14/12/22 15:24:54 INFO sqoop.Sqoop: Running Sqoop version: 1.4.5  
14/12/22 15:24:56 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.  
  
14/12/22 15:24:56 INFO tool.CodeGenTool: Beginning code generation  
14/12/22 15:24:58 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `emp` AS t LIMIT 1  
14/12/22 15:24:58 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `emp` AS t LIMIT 1  
  
14/12/22 15:24:58 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop  
14/12/22 15:25:11 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-  
hadoop/compile/cebe706d23ebb1fd99c1f063ad51ebd7/emp.jar  
-----  
14/12/22 15:25:40 INFO mapreduce.Job: The url to track the job:  
http://localhost:8088/proxy/application_1419242001831_0001/  
14/12/22 15:26:45 INFO mapreduce.Job: Job job_1419242001831_0001 running in uber mode :  
false  
14/12/22 15:26:45 INFO mapreduce.Job: map 0% reduce 0%  
14/12/22 15:28:08 INFO mapreduce.Job: map 100% reduce 0%  
14/12/22 15:28:16 INFO mapreduce.Job: Job job_1419242001831_0001 completed successfully  
-----  
14/12/22 15:28:17 INFO mapreduce.ImportJobBase: Transferred 145 bytes in 177.5849 seconds  
(0.8165 bytes/sec)  
14/12/22 15:28:17 INFO mapreduce.ImportJobBase: Retrieved 5 records.
```

To verify the imported data in HDFS, use the following command.

```
$ $HADOOP_HOME/bin/hadoop fs -cat /emp/part-m-*
```

It shows you the **emp** table data and fields are separated with comma , .

1201, gopal,	manager, 50000, TP
1202, manisha,	preader, 50000, TP
1203, kalil,	php dev, 30000, AC
1204, prasanth,	php dev, 30000, AC
1205, kranthi,	admin, 20000, TP

Importing into Target Directory

We can specify the target directory while importing table data into HDFS using the Sqoop import tool.

Following is the syntax to specify the target directory as option to the Sqoop import command.

```
--target-dir <new or exist directory in HDFS>
```

The following command is used to import **emp_add** table data into „queryresult“ directory.

```
$ sqoop import \  
--connect jdbc:mysql://localhost/userdb \  
--username root \  
--table emp_add \  
--m 1 \  
--target-dir /queryresult
```

The following command is used to verify the imported data in /queryresult directory from **emp_add** table.

```
$ $HADOOP_HOME/bin/hadoop fs -cat /queryresult/part-m-*
```

It will show you the emp_add table data with comma , separated fields.

```
1201, 288A, vgiri, jublee  
1202, 108I, aoc, sec-bad  
1203, 144Z, pgutta, hyd  
1204, 78B, oldcity, sec-bad  
1205, 720C, hitech, sec-bad
```

Import Subset of Table Data

We can import a subset of a table using the „where“ clause in Sqoop import tool. It executes the corresponding SQL query in the respective database server and stores the result in a target directory in HDFS.

The syntax for where clause is as follows.

```
--where <condition>
```

The following command is used to import a subset of **emp_add** table data. The subset query is to retrieve the employee id and address, who lives in Secunderabad city.

```
$ sqoop import \  
--connect jdbc:mysql://localhost/userdb \  
--username root \  
--table emp_add \  
--m 1 \  
--where "city = 'sec-bad'" \  
--target-dir /wherequery
```

The following command is used to verify the imported data in /wherequery directory from the **emp_add** table.

```
$ $HADOOP_HOME/bin/hadoop fs -cat /wherequery/part-m-*
```

It will show you the **emp_add** table data with comma , separated fields.

```
1202, 108I, aoc, sec-bad  
1204, 78B, oldcity, sec-bad  
1205, 720C, hitech, sec-bad
```

Incremental Import

Incremental import is a technique that imports only the newly added rows in a table. It is required to add „incremental“, „check-column“, and „last-value“ options to perform the incremental import.

The following syntax is used for the incremental option in Sqoop import command.

```
--incremental <mode>
```

```
--check-column <column name>
--last value <last check column value>
```

Let us assume the newly added data into emp table is as follows:

```
1206, satish p, grp des, 20000, GR
```

The following command is used to perform the incremental import in the **emp** table.

```
$ sqoop import \
--connect jdbc:mysql://localhost/userdb \
--username root \
--table emp \
--m 1 \
--incremental append \
--check-column id \
--last value 1205
```

The following command is used to verify the imported data from emp table to HDFS emp/ directory.

```
$ $HADOOP_HOME/bin/hadoop fs -cat /emp/part-m-*
```

It shows you the **emp** table data with comma , separated fields.

```
1201, gopal,      manager, 50000, TP
1202, manisha,    preader, 50000, TP
1203, kalil,      php dev, 30000, AC
1204, prasanth,   php dev, 30000, AC
1205, kranthi,    admin,    20000, TP
1206, satish p,   grp des,  20000, GR
```

The following command is used to see the modified or newly added rows from the **emp** table.

```
$ $HADOOP_HOME/bin/hadoop fs -cat /emp/part-m-*1
```

It shows you the newly added rows to the **emp** table with comma , separated fields.

```
1206, satish p, grp des, 20000, GR
```

Loading [MathJax]/jax/output/HTML-CSS/jax.js