

# Java-Success.com

Prepare to fast-track, choose & go places with 800+ Java & Big Data Q&As with lots of code & diagrams.

[Home](#) [Why? ▾](#) [300+ Java FAQs ▾](#) [300+ Big Data FAQs ▾](#) [Courses ▾](#)

[Membership ▾](#) [Your Career ▾](#)

[Home](#) › [bigdata-success.com](#) › [Tutorials - Big Data](#) › [TUT - Cloudera on Docker](#) › 14:

Docker Tutorial: Hive (via beeline) on Cloudera quickstart

## 14: Docker Tutorial: Hive (via beeline) on Cloudera quickstart

 Posted on [May 30, 2019](#)

This extends [Docker Tutorial: BigData on Cloudera quickstart via Docker](#).

**Step 1:** Run the container on a command line.

```
1 ~/projects/docker-hadoop]$ docker run --hostname=q
2 --privileged=true -t -i -v /Users/arulkumarankumar
3 --publish-all=true -p 8888:8888 -p 80:80 -p 7180:71
```

**Step 2:** Create a csv file “employee.csv” in the local file system.

### 300+ Java Interview FAQs

300+ Java FAQs



16+ Java Key Areas Q&As



150+ Java Architect FAQs



80+ Java Code Quality Q&As



150+ Java Coding Q&As



### 300+ Big Data Interview FAQs

300+ Big Data FAQs



Tutorials - Big Data



TUT -  Starting Big Data

TUT - Starting Spark & Scala

```
1 [root@quickstart /]# touch employee.csv
2 [root@quickstart /]# vi employee.csv
3
```

```
1 John, Samuel, IT
2 Peter, Smith, Finance
3 Sean, Mendis, Marketing
```

**Step 3:** Copy this file onto HDFS file system.

```
1 [root@quickstart /]# hdfs dfs -copyFromLocal employ
```

List the file in HDFS:

```
1 [root@quickstart /]# hdfs dfs -ls hdfs:///user/root
2 Found 1 items
3 -rw-r--r--    1 root supergroup          63 2019-05-31
4 [root@quickstart /]#
```

**Step 4:** Connect to hive 2 via **beeline**. Beeline is a thin client that also uses the Hive JDBC driver but instead executes queries through HiveServer2, which allows multiple concurrent client connections and supports authentication.

## Connect

```
1 [root@quickstart /]# beeline
2 ....
3 beeline>!connect jdbc:hive2://quickstart.cloudera:
4 .....
5
```

## Create a database

```
1 0: jdbc:hive2://quickstart.cloudera:10000> create
2 .....
```

TUT - Starting with Python

TUT - Kafka

TUT - Pig

TUT - Apache Storm

TUT - Spark Scala on Zeppelin

TUT - Cloudera

TUT - Cloudera on Docker

TUT - File Formats

TUT - Spark on Docker

TUT - Flume

TUT - Hadoop (HDFS)

TUT - HBase (NoSQL)

TUT - Hive (SQL)

TUT - Hadoop & Spark

TUT - MapReduce

TUT - Spark and Scala

TUT - Spark & Java

TUT - PySpark on Databricks

TUT - Zookeeper

## 800+ Java Interview Q&As

300+ Core Java Q&As



300+ Enterprise Java Q&As



150+ Java Frameworks Q&As



120+ Companion Tech Q&As



Tutorials - Enterprise Java



3

## Show databases

```
1 0: jdbc:hive2://quickstart.cloudera:10000> show databases;
2 .....
3 INFO : OK
4 +-----+
5 | database_name |
6 +-----+
7 | default      |
8 | mydb         |
9 | test         |
10 +-----+
11 3 rows selected (0.358 seconds)
12
```

## Create an external table

An external table created over a file in HDFS will not delete the file even if the table is dropped.

```
1 CREATE EXTERNAL TABLE IF NOT EXISTS mydb.tbl_employ
2 (first_name STRING, surname STRING, department STRIN
3 ROW FORMAT DELIMITED
4 FIELDS TERMINATED BY ','
5 STORED AS TEXTFILE
6 LOCATION 'hdfs://quickstart.cloudera:8020/user/root
7
```

## List the tables

```
1 0: jdbc:hive2://quickstart.cloudera:10000> use mydb;
2 ...
3 0: jdbc:hive2://quickstart.cloudera:10000> show tables;
4 ....
5 +-----+
6 | tab_name |
7 +-----+
8 | tbl_employee |
9 +-----+
10 1 row selected (0.036 seconds)
11
```

## SELECT \*

```

1 0: jdbc:hive2://quickstart.cloudera:10000> select
2 ....
3 +-----+-----+
4 | tbl_employee.first_name | tbl_employee.surname
5 +-----+-----+
6 | John                    | Samuel
7 | Peter                   | Smith
8 | Sean                    | Mendis
9 +-----+-----+
10

```

## INSERT INTO

```

1 0: jdbc:hive2://quickstart.cloudera:10000> INSERT INTO
2 0: jdbc:hive2://quickstart.cloudera:10000>   VALUES
3

```

```

1 0: jdbc:hive2://quickstart.cloudera:10000> SELECT
2 ....
3 +-----+-----+
4 | tbl_employee.first_name | tbl_employee.surname
5 +-----+-----+
6 | Rebecca                | Smith
7 | John                   | Samuel
8 | Peter                   | Smith
9 | Sean                    | Mendis
10 +-----+-----+
11

```

Exit out of beeline with **ctrl+z**.

```

1 [root@quickstart /]# hdfs dfs -ls hdfs://quickstart/
2 Found 2 items
3 -rwxrwxrwx   1 scott supergroup      20 2019-05-14 14:24
4 -rw-r--r--   1 root  supergroup     63 2019-05-14 14:24
5 [root@quickstart /]#

```

```

1 [root@quickstart /]# hdfs dfs -cat hdfs://quickstart/
2 Rebecca,Smith,Sales
3 [root@quickstart /]#

```

4

## Impala Shell

```
1 [root@quickstart /]# impala-shell
2 ....
3 [quickstart.cloudera:21000] > show databases;
4 Query: show databases
5 +-----+-----+
6 | name          | comment
7 +-----+-----+
8 | _impala_builtins | System database for Impala bu
9 | default        | Default Hive database
10 +-----+-----+
11 Fetched 2 row(s) in 0.13s
12 [quickstart.cloudera:21000] >
13
```

Why isn't "mydb" created with hive2 not showing even though both Impala & Hive 2 share the same metastore?

You need to run "INVALIDATE METADATA;" to be able to view the databases added via Hive 2.

```
1 [quickstart.cloudera:21000] > INVALIDATE METADATA
2 Query: invalidate METADATA
3
4 Fetched 0 row(s) in 4.61s
5 [quickstart.cloudera:21000] > show databases;
6 Query: show databases
7 +-----+-----+
8 | name          | comment
9 +-----+-----+
10 | _impala_builtins | System database for Impala bu
11 | default        | Default Hive database
12 | mydb           |
13 +-----+-----+
14 Fetched 3 row(s) in 0.00s
15 [quickstart.cloudera:21000] >
16
```

## Output beeline query to csv file

```
1 [root@quickstart /]# beeline -u jdbc:hive2://quickstart
2 --outputformat=csv2 -e "SELECT * FROM mydb.tbl_employee"
3
4
5
6
7
```

```
1 [root@quickstart /]# cat output.csv
2 tbl_employee.first_name,tbl_employee.surname,tbl_employee.department
3 John, Samuel, IT
4 Peter, Smith, Finance
5 Sean, Mendis, Marketing
6 [root@quickstart /]#
7
```

◀ 13: Docker Tutorial: Apache Spark (spark-shell & pyspark) on Cloudera

quickstart

15: Docker Tutorial: Hive & parquet-tools – csv to parquet on Cloudera

quickstart ▶

## Disclaimer

The contents in this Java-Success are copyrighted and from EmpoweringTech pty Ltd. The EmpoweringTech pty Ltd has the right to correct or enhance the current content without any prior notice. These are general advice only, and one needs to take his/her own circumstances into consideration. The EmpoweringTech pty Ltd will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. Links to external sites do not imply endorsement of the linked-to sites. [Privacy Policy](#).