

# Java-Success.com

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

search here ...

Go

- Home
- Why? ▾
- 300+ Java FAQs ▾
- 300+ Big Data FAQs ▾
- Courses ▾
- Membership ▾
- Your Career ▾

Home > bigdata-success.com > Tutorials - Big Data > TUT - Cloudera on Docker > 11:

Docker Tutorial: Hadoop Big Data CLIs on Cloudera quickstart

## 11: Docker Tutorial: Hadoop Big Data CLIs on Cloudera quickstart


 Posted on May 26, 2019

There are a number of CLIs that you can run from the edge node, which is the gateway node to the Hadoop cluster consisting of master & slave (aka worker) nodes. Let's look at the different **CLIs** (i.e **Command Line Interfaces**)



Most of the CLIs listed below are in **/usr/bin**

|   |                                                   |   |      |      |      |        |      |    |  |  |
|---|---------------------------------------------------|---|------|------|------|--------|------|----|--|--|
| 1 | [root@quickstart /]# ls -ltr /usr/bin   grep 'hd' |   |      |      |      |        |      |    |  |  |
| 2 | -rwxr-xr-x                                        | 1 | root | root | 8144 | Sep 22 | 2015 | df |  |  |
| 3 | -rwxr-xr-x                                        | 1 | root | root | 328  | Mar 23 | 2016 | h  |  |  |
| 4 | -rwxr-xr-x                                        | 1 | root | root | 321  | Mar 23 | 2016 | h  |  |  |
| 5 | -rwxr-xr-x                                        | 1 | root | root | 324  | Mar 23 | 2016 | b  |  |  |

### 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

|    |            |   |      |      |         |        |      |    |
|----|------------|---|------|------|---------|--------|------|----|
| 6  | -rwxr-xr-x | 1 | root | root | 504     | Mar 23 | 2016 | sc |
| 7  | -rwxr-xr-x | 1 | root | root | 1138056 | Mar 23 | 2016 | g  |
| 8  | -rwxr-xr-x | 1 | root | root | 141     | Mar 23 | 2016 | sp |
| 9  | -rwxr-xr-x | 1 | root | root | 140     | Mar 23 | 2016 | sp |
| 10 | -rwxr-xr-x | 1 | root | root | 185     | Mar 23 | 2016 | py |
| 11 | -rwxr-xr-x | 1 | root | root | 192     | Mar 23 | 2016 | ha |
| 12 | -rwxr-xr-x | 1 | root | root | 1856    | Mar 23 | 2016 | ir |
| 13 |            |   |      |      |         |        |      |    |

## hdfs CLI

The following command will give you the commands you can use with “hdfs”

```
1 [root@quickstart /]# hdfs
```

“dfs” – run a filesystem command on the file systems supported in Hadoop.

|    |                                         |   |          |            |   |      |  |  |
|----|-----------------------------------------|---|----------|------------|---|------|--|--|
| 1  | [root@quickstart /]# hdfs dfs -ls /user |   |          |            |   |      |  |  |
| 2  | Found 8 items                           |   |          |            |   |      |  |  |
| 3  | drwxr-xr-x                              | - | cloudera | cloudera   | 0 | 2016 |  |  |
| 4  | drwxr-xr-x                              | - | mapred   | hadoop     | 0 | 2016 |  |  |
| 5  | drwxrwxrwx                              | - | hive     | supergroup | 0 | 2016 |  |  |
| 6  | drwxrwxrwx                              | - | hue      | supergroup | 0 | 2016 |  |  |
| 7  | drwxrwxrwx                              | - | jenkins  | supergroup | 0 | 2016 |  |  |
| 8  | drwxrwxrwx                              | - | oozie    | supergroup | 0 | 2016 |  |  |
| 9  | drwxrwxrwx                              | - | root     | supergroup | 0 | 2016 |  |  |
| 10 | drwxr-xr-x                              | - | hdfs     | supergroup | 0 | 2016 |  |  |

You can list all the COMMANDS via

```
1 [root@quickstart /]# hdfs dfs
```

This is a very handy utility that lets you copy files between local Unix & Hadoop file systems.

## hadoop CLI

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



**fs** refers to any file system, it could be local or HDFS  
but **dfs** shown above refers to only HDFS file system.

## fs

Run a generic filesystem user client.

## HDFS file system

```
1 [root@quickstart /]# hadoop fs -ls hdfs:///
2 Found 5 items
3 drwxrwxrwx - hdfs supergroup 0 2016-04-
4 drwxr-xr-x - hbase supergroup 0 2019-06-
5 drwxrwxrwt - hdfs supergroup 0 2019-06-
6 drwxr-xr-x - hdfs supergroup 0 2016-04-
7 drwxr-xr-x - hdfs supergroup 0 2016-04-
8
```

## Unix file system

```
1 [root@quickstart /]# hadoop fs -ls file:///
2 Found 24 items
3 -rwxr-xr-x 1 root root 0 2019-06-08 11
4 -rwxr-xr-x 1 root root 0 2016-04-06 02
5 dr-xr-xr-x - root root 4096 2016-04-06 02
6 drwxr-xr-x - root root 4096 2016-04-06 02
7 drwxr-xr-x - root root 2920 2019-06-08 11
8 drwxr-xr-x - root root 4096 2019-06-08 11
9 drwxrwxr-x - root root 4096 2016-04-05 23
10 dr-xr-xr-x - root root 4096 2016-04-06 02
11 dr-xr-xr-x - root root 12288 2016-04-06 02
12 drwx----- - root root 4096 2015-03-04 12
13 drwxr-xr-x - root root 4096 2011-09-23 11
14 drwxr-xr-x - root root 4096 2011-09-23 11
15 drwxr-xr-x - root root 4096 2016-04-06 02
16 drwxr-xr-x - root root 4096 2016-04-06 02
17 dr-xr-xr-x - root root 0 2019-06-08 11
18 dr-xr-x-- - root root 4096 2016-04-06 02
19 dr-xr-xr-x - root root 4096 2016-04-06 02
20 drwxr-xr-x - root root 4096 2015-03-04 12
21 drwxr-xr-x - root root 352 2019-05-21 12
22 drwxr-xr-x - root root 4096 2011-09-23 11
23 dr-xr-xr-x - root root 0 2019-06-08 11
24 drwxrwxrwt - root root 4096 2019-06-08 11
25 drwxrwxr-x - root root 4096 2016-04-06 02
26 drwxr-xr-x - root root 4096 2016-04-06 02
27 [root@quickstart /]#
28
```

## distcp

copy file or directories recursively. It is very useful in bulk copying files between local & hdfs file systems. It basically runs a **MapReduce** job.

Copy a Unix file system file to HDFS:

```
1 [root@quickstart /]# hadoop distcp file:///var/log
2 .....
3 .....
4 19/06/08 11:50:32 INFO tools.DistCp: DistCp job-id
5 19/06/08 11:50:32 INFO mapreduce.Job: Running job
6 19/06/08 11:50:37 INFO mapreduce.Job: Job job_1559
7 19/06/08 11:50:37 INFO mapreduce.Job: map 0% redu
8 19/06/08 11:50:42 INFO mapreduce.Job: map 100% re
9 19/06/08 11:50:42 INFO mapreduce.Job: Job job_1559
10 19/06/08 11:50:42 INFO mapreduce.Job: Counters: 33
11
1 [root@quickstart /]# hadoop fs -ls hdfs:///user/lo
2 -rw-r--r--  1 root supergroup      146584 2019-06-0
3 [root@quickstart /]#
4
```

You can even copy files between AWS s3 buckets & HDFS using distcp.

```
1 $ hadoop distcp s3a://bucket/file hdfs://path/to/f
2
```

## Hive beeline CLI

```
1 [root@quickstart /]# hive
2 CREATE DATABASE IF NOT EXISTS test;
3 OK
4 Time taken: 0.301 seconds
5 hive> SHOW DATABASES;
6 OK
7 default
8 test
```

```
9 | Time taken: 0.24 seconds, Fetched: 2 row(s)
10 |
```

The Hive CLI connects directly to the Hive Driver and requires that Hive be installed on the same machine as the client. However, Beeline connects to HiveServer2 and does not require the installation of Hive libraries on the same machine as the client. 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.

```
1 | [root@quickstart /]# beeline
2 | beeline> !connect jdbc:hive2://localhost:10000 org
3 | scan complete in 2ms
4 | Connecting to jdbc:hive2://localhost:10000
5 | Enter password for jdbc:hive2://localhost:10000:
6 |
```

Just press “enter” when prompted for the password as no password is set up. In real enterprise systems kerberos based security will be enabled and you need to provide the path to the keytab file.

```
1 | Connected to: Apache Hive (version 1.1.0-cdh5.7.0)
2 | Driver: Hive JDBC (version 1.1.0-cdh5.7.0)
3 | Transaction isolation: TRANSACTION_REPEATABLE_READ
4 | 0: jdbc:hive2://localhost:10000> SHOW DATABASES;
5 | INFO : Compiling command(queryId=hive_2019052504)
6 | INFO : Semantic Analysis Completed
7 | INFO : Returning Hive schema: Schema(fieldSchemas:
8 | INFO : Completed compiling command(queryId=hive_2
9 | INFO : Concurrency mode is disabled, not creating
10 | INFO : Executing command(queryId=hive_2019052504)
11 | INFO : Starting task [Stage-0:DDL] in serial mode
12 | INFO : Completed executing command(queryId=hive_2
13 | INFO : OK
14 | +-----+-----+
15 | | database_name |
16 | +-----+-----+
17 | | default      |
18 | | test        |
```

```

19 +-----+
20 2 rows selected (0.125 seconds)
21 0: jdbc:hive2://localhost:10000>
22

```

## Impala CLI

```

1 [root@quickstart /]# impala-shell
2 Starting Impala Shell without Kerberos authentication
3 Connected to quickstart.cloudera:21000
4 Server version: impalad version 2.5.0-cdh5.7.0 RELEASE
5 *****
6 Welcome to the Impala shell. Copyright (c) 2015 Cloudera, Inc.
7 (Impala Shell v2.5.0-cdh5.7.0 (ad3f5ad) built on Wed Aug 12 2015)
8
9 Want to know what version of Impala you're connected to? Type
10 find out!
11 *****
12 [quickstart.cloudera:21000] > SHOW DATABASES;
13 Query: show DATABASES
14 +-----+
15 | name          | comment
16 +-----+
17 | _impala_builtins | System database for Impala builtins
18 | default        | Default Hive database
19 +-----+
20 Fetched 2 row(s) in 0.13s
21 [quickstart.cloudera:21000] >
22

```

When you create a database in Impala, the database can also be used by Hive. When you create a database in Hive, issue an **INVALIDATE METADATA** statement in Impala to make Impala permanently aware of the new database.

```

1 [quickstart.cloudera:21000] > INVALIDATE METADATA
2 Query: invalidate METADATA
3
4 Fetched 0 row(s) in 3.31s
5 [quickstart.cloudera:21000] > SHOW DATABASES;
6 Query: show DATABASES
7 +-----+
8 | name          | comment
9 +-----+
10 | _impala_builtins | System database for Impala builtins
11 | default        | Default Hive database

```

```
12 | test |
13 +-----+
14 Fetched 3 row(s) in 0.01s
15 [quickstart.cloudera:21000] >
16
```

The **Hive metastore** service stores the metadata for Hive tables and partitions in a relational database. Impala keeps its table definitions in a traditional MySQL or PostgreSQL database known as the metastore, the same database where Hive keeps this type of data.

## Hadoop YARN CLI

Apache Hadoop YARN decentralizes execution and monitoring of processing jobs. In a clustered architecture, Apache Hadoop YARN sits between HDFS and the processing (E.g. Hive, Impala, Spark-submit, etc) engines being used to run applications. It combines a central resource manager with containers, application coordinators and node-level agents that monitor processing operations in individual cluster nodes. YARN can dynamically allocate resources to applications as needed, a capability designed to improve resource utilization and application performance compared with MapReduce's more static allocation approach.

```
1 [root@quickstart /]# yarn
```

You can list the running applications, kill, inspect the states as in ALL, NEW, NEW\_SAVING, SUBMITTED, ACCEPTED, RUNNING, FINISHED, FAILED, KILLED, etc.

```
1 [root@quickstart /]# yarn application -list
```

```
2 19/05/25 04:39:52 INFO client.RMPProxy: Connecting
3 Total number of applications (application-types: [
4 Application-Id Application-Nam
5 [root@quickstart /]#
```

# Spark-shell CLI

You can run your Spark application interactively in Scala.

```

1 [root@quickstart ~]# spark-shell
2 SLF4J: Class path contains multiple SLF4J bindings:
3 SLF4J: Found binding in [jar:file:/usr/lib/zookeeper-
4 SLF4J: Found binding in [jar:file:/usr/jars/slf4j-
5 SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings
6 SLF4J: Actual binding is of type [org.slf4j.impl.Log4jSLF4JImpl]
7 Setting default log level to "WARN".
8 To adjust logging level use sc.setLogLevel(newLevel).
9 Welcome to
10
11      ____
12     /  __ \   _ __   ____
13    /  /  \  /  __ \  / __ \
14   /  /  \ /  /  \ /  / __ \
15  /  /  \ /  /  \ /  / __ \
16  /  /  \ /  /  \ /  / __ \
17  /  /  \ /  /  \ /  / __ \
18  /  /  \ /  /  \ /  / __ \
19  /  /  \ /  /  \ /  / __ \
20  /  /  \ /  /  \ /  / __ \
21  /  /  \ /  /  \ /  / __ \
22  /  /  \ /  /  \ /  / __ \
23  /  /  \ /  /  \ /  / __ \
24  /  /  \ /  /  \ /  / __ \
25  /  /  \ /  /  \ /  / __ \
26  /  /  \ /  /  \ /  / __ \
27  /  /  \ /  /  \ /  / __ \
28  /  /  \ /  /  \ /  / __ \
29  /  /  \ /  /  \ /  / __ \
30  /  /  \ /  /  \ /  / __ \
31  /  /  \ /  /  \ /  / __ \
32  /  /  \ /  /  \ /  / __ \
33  /  /  \ /  /  \ /  / __ \
34  /  /  \ /  /  \ /  / __ \
35  /  /  \ /  /  \ /  / __ \
36  /  /  \ /  /  \ /  / __ \
37  /  /  \ /  /  \ /  / __ \
38  /  /  \ /  /  \ /  / __ \
39  /  /  \ /  /  \ /  / __ \
40  /  /  \ /  /  \ /  / __ \
41  /  /  \ /  /  \ /  / __ \
42  /  /  \ /  /  \ /  / __ \
43  /  /  \ /  /  \ /  / __ \
44  /  /  \ /  /  \ /  / __ \
45  /  /  \ /  /  \ /  / __ \
46  /  /  \ /  /  \ /  / __ \
47  /  /  \ /  /  \ /  / __ \
48  /  /  \ /  /  \ /  / __ \
49  /  /  \ /  /  \ /  / __ \
50  /  /  \ /  /  \ /  / __ \
51  /  /  \ /  /  \ /  / __ \
52  /  /  \ /  /  \ /  / __ \
53  /  /  \ /  /  \ /  / __ \
54  /  /  \ /  /  \ /  / __ \
55  /  /  \ /  /  \ /  / __ \
56  /  /  \ /  /  \ /  / __ \
57  /  /  \ /  /  \ /  / __ \
58  /  /  \ /  /  \ /  / __ \
59  /  /  \ /  /  \ /  / __ \
60  /  /  \ /  /  \ /  / __ \
61  /  /  \ /  /  \ /  / __ \
62  /  /  \ /  /  \ /  / __ \
63  /  /  \ /  /  \ /  / __ \
64  /  /  \ /  /  \ /  / __ \
65  /  /  \ /  /  \ /  / __ \
66  /  /  \ /  /  \ /  / __ \
67  /  /  \ /  /  \ /  / __ \
68  /  /  \ /  /  \ /  / __ \
69  /  /  \ /  /  \ /  / __ \
70  /  /  \ /  /  \ /  / __ \
71  /  /  \ /  /  \ /  / __ \
72  /  /  \ /  /  \ /  / __ \
73  /  /  \ /  /  \ /  / __ \
74  /  /  \ /  /  \ /  / __ \
75  /  /  \ /  /  \ /  / __ \
76  /  /  \ /  /  \ /  / __ \
77  /  /  \ /  /  \ /  / __ \
78  /  /  \ /  /  \ /  / __ \
79  /  /  \ /  /  \ /  / __ \
80  /  /  \ /  /  \ /  / __ \
81  /  /  \ /  /  \ /  / __ \
82  /  /  \ /  /  \ /  / __ \
83  /  /  \ /  /  \ /  / __ \
84  /  /  \ /  /  \ /  / __ \
85  /  /  \ /  /  \ /  / __ \
86  /  /  \ /  /  \ /  / __ \
87  /  /  \ /  /  \ /  / __ \
88  /  /  \ /  /  \ /  / __ \
89  /  /  \ /  /  \ /  / __ \
90  /  /  \ /  /  \ /  / __ \
91  /  /  \ /  /  \ /  / __ \
92  /  /  \ /  /  \ /  / __ \
93  /  /  \ /  /  \ /  / __ \
94  /  /  \ /  /  \ /  / __ \
95  /  /  \ /  /  \ /  / __ \
96  /  /  \ /  /  \ /  / __ \
97  /  /  \ /  /  \ /  / __ \
98  /  /  \ /  /  \ /  / __ \
99  /  /  \ /  /  \ /  / __ \
100 /  /  \ /  /  \ /  / __ \

```

## Pyspark CLI

You can run your Spark application interactively in Python.



```

1 [root@quickstart /]# pyspark
2 Python 2.6.6 (r266:84292, Jul 23 2015, 15:22:56)
3 [GCC 4.4.7 20120313 (Red Hat 4.4.7-11)] on linux2
4 Type "help", "copyright", "credits" or "license()"
5 SLF4J: Class path contains multiple SLF4J bindings:
6 SLF4J: Found binding in [jar:file:/usr/lib/zookeep
7 SLF4J: Found binding in [jar:file:/usr/jars/slf4j-
8 SLF4J: See http://www.slf4j.org/codes.html#multip
9 SLF4J: Actual binding is of type [org.slf4j.impl.L
10 19/05/25 15:06:01 INFO spark.SparkContext: Running
11 19/05/25 15:06:01 WARN util.NativeCodeLoader: Unab
12 19/05/25 15:06:02 INFO spark.SecurityManager: Char
13 19/05/25 15:06:02 INFO spark.SecurityManager: Char
14 19/05/25 15:06:02 INFO spark.SecurityManager: Secu
15 19/05/25 15:06:02 INFO util.Utills: Successfully st
16 19/05/25 15:06:02 INFO slf4j.Slf4jLogger: Slf4jLog
17 19/05/25 15:06:02 INFO Remoting: Starting remoting
18 19/05/25 15:06:02 INFO Remoting: Remoting started
19 19/05/25 15:06:02 INFO Remoting: Remoting now list
20 19/05/25 15:06:02 INFO util.Utills: Successfully st
21 19/05/25 15:06:02 INFO spark.SparkEnv: Registering
22 19/05/25 15:06:02 INFO spark.SparkEnv: Registering
23 19/05/25 15:06:02 INFO storage.DiskBlockManager: (
24 19/05/25 15:06:02 INFO storage.MemoryStore: Memory
25 19/05/25 15:06:02 INFO spark.SparkEnv: Registering
26 19/05/25 15:06:02 INFO server.Server: jetty-8.y.z-
27 19/05/25 15:06:03 INFO server.AbstractConnector: S
28 19/05/25 15:06:03 INFO util.Utills: Successfully st
29 19/05/25 15:06:03 INFO ui.SparkUI: Started SparkUI
30 19/05/25 15:06:03 INFO executor.Executor: Starting
31 19/05/25 15:06:03 INFO util.Utills: Successfully st
32 19/05/25 15:06:03 INFO netty.NettyBlockTransferSer
33 19/05/25 15:06:03 INFO storage.BlockManagerMaster
34 19/05/25 15:06:03 INFO storage.BlockManagerMasterI
35 19/05/25 15:06:03 INFO storage.BlockManagerMaster
36 Welcome to
37
38      _ _ _ _ _
39     / _ \ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
40    / _ \ / _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ version 1.6.0
41     / _ \
42
43 Using Python version 2.6.6 (r266:84292, Jul 23 20
44 SparkContext available as sc, HiveContext availab
45 >>>
46

```

## Spark-submit CLI

spark-shell & pyspark are for interactive queries, and need to be run in **yarn-client** mode so that the

machine you're running on acts as the driver. Client mode is for learning purpose.

spark-submit is for submitting the jobs to the cluster where the task runs in the cluster. Production jobs run in **cluster mode** so that YARN can assign the driver to a suitable node on the cluster with available resources.

## Do you want to open multiple terminal windows?

|   |                                           |                     |           |
|---|-------------------------------------------|---------------------|-----------|
| 1 | \$ docker ps                              |                     |           |
| 2 | CONTAINER ID                              | IMAGE               | COMMAND   |
| 3 | bf645c6a2930                              | cloudera/quickstart | "/usr/bin |
| 4 |                                           |                     |           |
|   |                                           |                     |           |
| 1 | \$ docker exec -it bf645c6a2930 /bin/bash |                     |           |
| 2 | [root@quickstart /]#                      |                     |           |
| 3 |                                           |                     |           |

## How to stop the container?

|   |                             |                     |           |
|---|-----------------------------|---------------------|-----------|
| 1 | \$ docker ps                |                     |           |
| 2 | CONTAINER ID                | IMAGE               | COMMAND   |
| 3 | bf645c6a2930                | cloudera/quickstart | "/usr/bin |
| 4 |                             |                     |           |
|   |                             |                     |           |
| 1 | \$ docker stop bf645c6a2930 |                     |           |
| 2 |                             |                     |           |

## What is next?

Look at the configuration files used in the Hadoop eco system.

◀ 10: Docker Tutorial: Hadoop Big Data services & folders on Cloudera quickstart

## 12: Docker Tutorial: Hadoop Big Data configuration files 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](#)

© 2022 [java-success.com](https://www.java-success.com)