

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](#) > 09:

Docker Tutorial: Getting started with Hadoop Big Data on Cloudera quickstart

09: Docker Tutorial: Getting started with Hadoop Big Data on Cloudera quickstart

 Posted on [May 24, 2019](#)

If you are not familiar with Docker get some hands-on experience at [a series of step by step Docker tutorials](#) with Java & Springboot examples.

Step 1: <https://hub.docker.com/> is a docker repository from where you can pull & push images. You can search for the images. You can see some of the below steps at:
<https://hub.docker.com/r/cloudera/quickstart>.

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

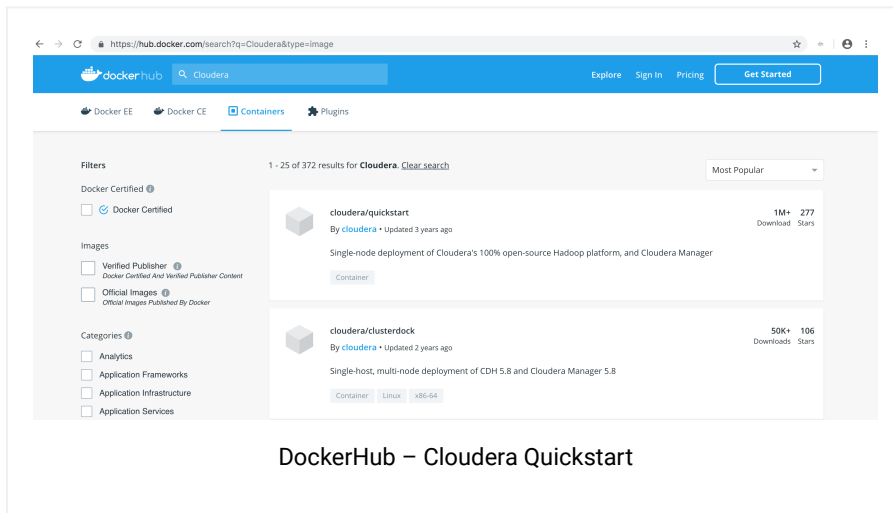


Tutorials - Big Data



TUT -  Starting Big Data

TUT - Starting Spark & Scala



```
1 ~/projects/docker-hadoop]$ docker pull cloudera/qu
```

It will take a few minutes to download.

Step 2: Check the downloaded image with

```
1 ~/projects/docker-hadoop]$ docker images
2 REPOSITORY          TAG          IMAGE ID
3 cloudera/quickstart  latest      4239cd295
```

Step 3: Create the container from the image “cloudera/quickstart”.

```
1 ~/projects/docker-hadoop]$ docker run --hostname=qu
```

Step 4: List the running containers and get the container id.

```
1 ~/projects/docker-hadoop]$ docker ps
2 CONTAINER ID        IMAGE          COMMAND
3 c6cbc91a69ad        cloudera/quickstart  "/usr/bin
```

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



You can see that 0.0.0.0:8888->8888/tcp, which means "0.0.0.0:8888" is the host ip address & port. You can also inspect the container by opening a new terminal and then type:

```
1 ~/projects/docker-hadoop]$ docker inspect 7d636c3a4
2
```

Where you can see:

```
1 ..... "NetworkSettings": {
2         "Bridge": "",
3         "SandboxID": "9f2fd6f8dc1b6a44b3da9384
4         "HairpinMode": false,
5         "LinkLocalIPv6Address": "",
6         "LinkLocalIPv6PrefixLen": 0,
7         "Ports": {
8             "7180/tcp": [
9                 {
10                    "HostIp": "0.0.0.0",
11                    "HostPort": "7180"
12                }
13            ],
14            "80/tcp": [
15                {
16                    "HostIp": "0.0.0.0",
17                    "HostPort": "80"
18                }
19            ],
20            "8888/tcp": [
21                {
22                    "HostIp": "0.0.0.0",
23                    "HostPort": "8888"
24                }
25            ]
26        },
27    }
```

HUE

HUE stands for Hadoop User Experience where you can browse the files in HDFS and run SQL like queries against Hive & Impala tables.

Step 5: Open a browser and type “http://0.0.0.0:8888” to open HUE GUI.

username: cloudera

password: cloudera

HDFS commands

You can run a number HDFS commands on a command-line as shown below. These commands are Unix like:

Step 6: On the original terminal you can practice the Hadoop commands like

```
1 [root@quickstart /]# hdfs dfs -ls /user
2 Found 9 items
3 drwxr-xr-x - cloudera cloudera 0 2016
4 drwxr-xr-x - hdfs supergroup 0 2019
5 drwxr-xr-x - mapred hadoop 0 2016
6 drwxrwxrwx - hive supergroup 0 2016
7 drwxrwxrwx - hue supergroup 0 2016
8 drwxrwxrwx - jenkins supergroup 0 2016
9 drwxrwxrwx - oozie supergroup 0 2016
10 drwxrwxrwx - root supergroup 0 2016
11 drwxr-xr-x - hdfs supergroup 0 2016
12
```

You can also see this via the HUE GUI by clicking on the “**File Browser**” at the top right. Click on the “/user” to see the same folders as above.

mkdir

```
1 [root@quickstart /]# hdfs dfs -mkdir /user/java-su
```

ls

```
1 [root@quickstart /]# hdfs dfs -ls /user
```

```
2 ound 10 items
3 drwxr-xr-x - cloudera cloudera 0 2016
4 drwxr-xr-x - hdfs supergroup 0 2019
5 drwxr-xr-x - mapred hadoop 0 2016
6 drwxrwxrwx - hive supergroup 0 2016
7 drwxrwxrwx - hue supergroup 0 2016
8 drwxr-xr-x - root supergroup 0 2019
9 drwxrwxrwx - jenkins supergroup 0 2016
10 drwxrwxrwx - oozie supergroup 0 2016
11 drwxrwxrwx - root supergroup 0 2016
12 drwxr-xr-x - hdfs supergroup 0 2016
13
```

touch

Create a file.

```
1 [root@quickstart /]# hdfs dfs -touchz /user/java-s
```

You can see what command options are available by typing:

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

Cloudera guide

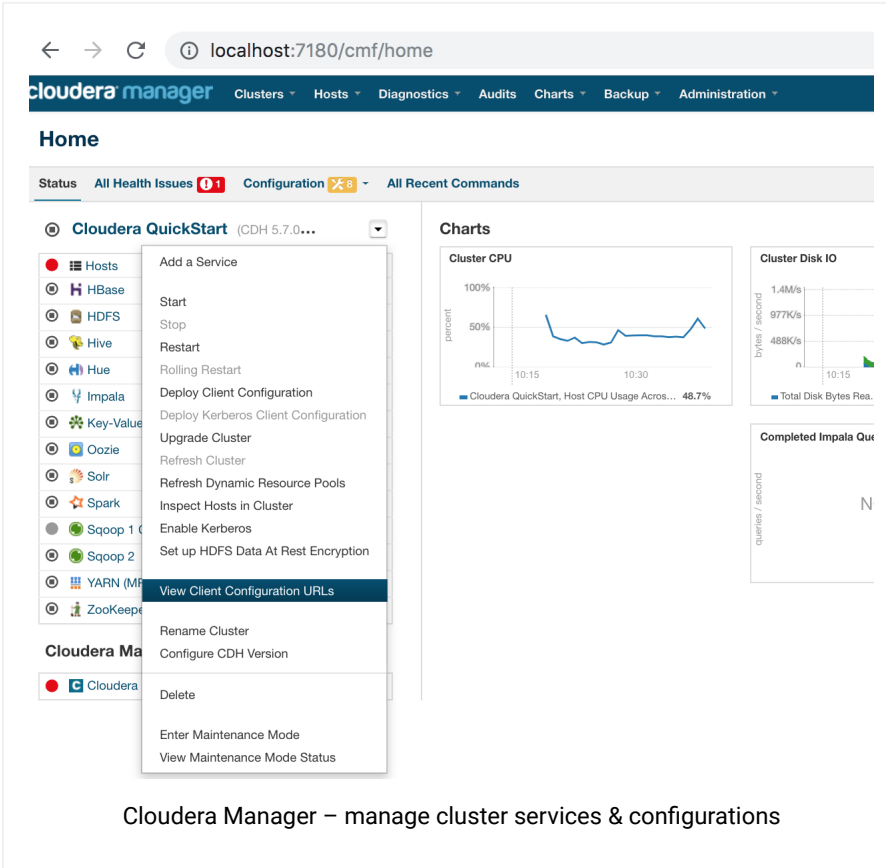
You can look at the guide & examples via:

“<http://localhost:80>” takes you to the Quickstart guide & tutorial.

Cloudera Manager

The Cloudera manager is not started by default. It requires at around 10 GB of RAM. Cloudera manager is a web UI to manage Hadoop cluster and services like Hive, Spark, Impala, HBase, etc. You can stop, start, and restart the services. You can modify the configuration values. You can monitor the jobs and their statuses.

```
1 [root@quickstart ~]# cd /home/cloudera/
2 [root@quickstart cloudera]# ./cloudera-manager --c
3 .....
4 -----
5
6 Success! You can now log into Cloudera Manager from
7
8 http://quickstart.cloudera:7180
9
10 Username: cloudera
11 Password: cloudera
12
13 [root@quickstart cloudera]#
14
```

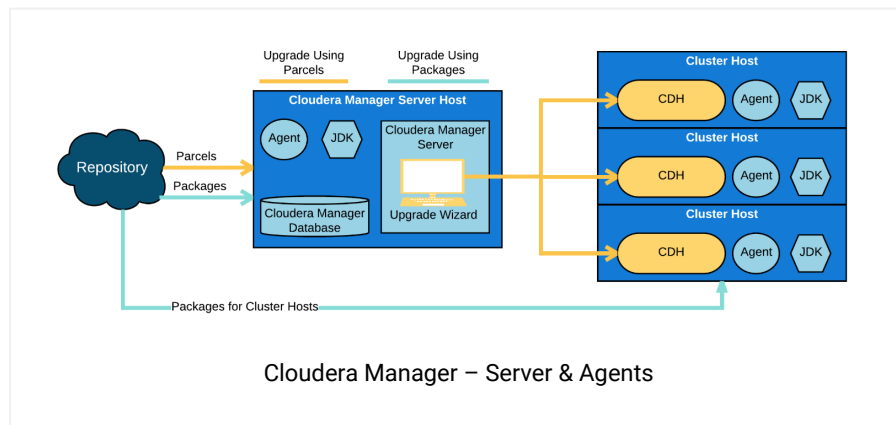


Cloudera Manager – manage cluster services & configurations

<http://localhost:7180/> to access the **Cloudera manager**. You can start all the services you require or only the services that you require, but it can consume lots of resources. Services may be in bad health due to lack of resources.

The Cloudera Manager Server is the master service that manages the data model of the entire cluster in a database. The data model contains information

regarding the services, roles, and configurations assigned for each node in the cluster. You can also upgrade the services via parcels & packages.



CDH – stands for **C**loudera **D**istribution **H**adoop. CDH upgrades contain updated versions of the Hadoop software and other components. You can use Cloudera Manager to upgrade CDH for major, minor, and maintenance upgrades.

Do you want to open multiple terminal windows?

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

| | | | |
|---|-----------------------------|--|--|
| 1 | \$ docker stop bf645c6a2930 | | |
|---|-----------------------------|--|--|

2

How to remove all exited containers?

List all containers inclusive of stopped containers:

```
1 $ docker ps -a
2
```

Remove all containers inclusive of stopped containers:

```
1 $ docker rm $(docker ps -a -q)
2
```

How to list all the images?

```
1 $ docker images -a
2 REPOSITORY          TAG          IMAGE ID
3 gdancik/cloudera     latest      49184101c
4 cloudera/quickstart  latest      4239cd295
5
```

What is next?

In the next tutorials will drill into Cloudera Quickstart – Services, CLIs, config files, etc to get a good overview. This compliments [Getting started with BigData on Cloudera](#), which was on a Virtual Machine.

These tutorials are based on lighter **Docker containers**. [10: Docker Tutorial: BigData services & folders on Cloudera quickstart](#).

◀ 08: Docker Tutorial – client & server with docker-compose.yml

10: Docker Tutorial: Hadoop Big Data services & folders 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)