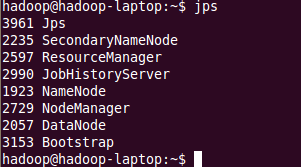
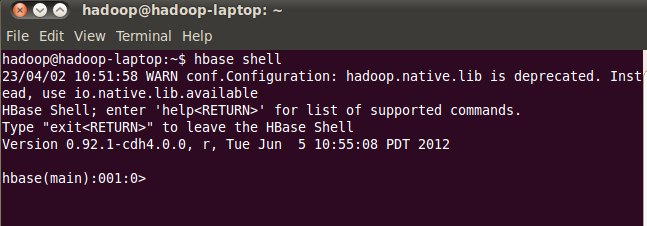
**Hbase commands**

Step 1:First go to terminal and type **StartCDH.sh**

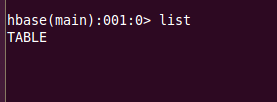
Step 2:Next type**jps** command in the terminal



Step 3:Type**hbase shell**



List will gives you list of tables in Hbase



Step 5:hbase(main):001:0>**version**



Version will gives you the version of hbase

### Create Table Syntax

CREATE 'name\_space:table\_name', 'column\_family’

**hbase(main):011:0> create 'newtbl','knowledge'**

**hbase(main):011:0>describe 'newtbl'**

**hbase(main):011:0>status**

**1 servers, 0 dead, 15.0000 average load**

# **HBase – Using PUT to Insert data to Table**

To insert data into the HBase table use PUT command, this would be similar to insert statement on RDBMS but the syntax is completely different. In this article I will describe how to insert data into HBase table with examples using PUT command from the HBase shell.

### HBase PUT command syntax

Below is the syntax of PUT command which is used to insert data (rows and columns) into a HBase table.

### HBase PUT command syntax

Below is the syntax of PUT command which is used to insert data (rows and columns) into a HBase table.

**put '<name\_space:table\_name>', '<row\_key>' '<cf:column\_name>', '<value>'**

hbase(main):015:0>**put 'newtbl','r1','knowledge:sports','cricket'**

0 row(s) in 0.0150 seconds

hbase(main):016:0>**put 'newtbl','r1','knowledge:science','chemistry'**

0 row(s) in 0.0040 seconds

hbase(main):017:0>**put 'newtbl','r1','knowledge:science','physics'**

0 row(s) in 0.0030 seconds

hbase(main):018:0>**put 'newtbl','r2','knowledge:economics','macroeconomics'**

0 row(s) in 0.0030 seconds

hbase(main):019:0>**put 'newtbl','r2','knowledge:music','songs'**

0 row(s) in 0.0170 seconds

hbase(main):020:0>**scan 'newtbl'**

ROW COLUMN+CELL

r1 column=knowledge:science, timestamp=1678807827189, value

=physics

r1 column=knowledge:sports, timestamp=1678807791753, value=

cricket

r2 column=knowledge:economics, timestamp=1678807854590, val

ue=macroeconomics

r2 column=knowledge:music, timestamp=1678807877340, value=s

ongs

2 row(s) in 0.0250 secondsdisable

To retrieve only the row1 data

hbase(main):023:0>**get 'newtbl', 'r1'**

**output**

COLUMN CELL

knowledge:science timestamp=1678807827189, value=physics

knowledge:sports timestamp=1678807791753, value=cricket

2 row(s) in 0.0150 seconds.

hbase(main):025:0> disable 'newtbl'

0 row(s) in 1.2760 seconds

### Verification

After disabling the table, you can still sense its existence through **list** and **exists** commands. You cannot scan it. It will give you the following error.

hbase(main):028:0> scan 'newtbl'

ROW COLUMN + CELL

ERROR: newtbl is disabled.

### is\_disabled

This command is used to find whether a table is disabled. Its syntax is as follows.

hbase>is\_disabled 'table name'

hbase(main):031:0>is\_disabled 'newtbl'

true

0 row(s) in 0.0440 seconds

### disable\_all

This command is used to disable all the tables matching the given regex. The syntax for **disable\_all** command is given below.

hbase>disable\_all 'r.\*'

Suppose there are 5 tables in HBase, namely raja, rajani, rajendra, rajesh, and raju. The following code will disable all the tables starting with **raj.**

hbase(main):002:07>disable\_all 'raj.\*'

raja

rajani

rajendra

rajesh

raju

Disable the above 5 tables (y/n)?

y

5 tables successfully disabled

## **Enabling a Table using HBase Shell**

Syntax to enable a table:

**enable ‘newtbl’**

### Example

Given below is an example to enable a table.

**hbase(main):005:0> enable 'newtbl'**

**0 row(s) in 0.4580 seconds**

### Verification

After enabling the table, scan it. If you can see the schema, your table is successfully enabled.

**hbase(main):006:0> scan 'newtbl'**

## **is\_enabled**

This command is used to find whether a table is enabled. Its syntax is as follows:

**hbase>is\_enabled 'table name'**

The following code verifies whether the table named **emp** is enabled. If it is enabled, it will return true and if not, it will return false.

hbase(main):031:0>is\_enabled 'newtbl'

true

0 row(s) in 0.0440 seconds

## **describe**

This command returns the description of the table. Its syntax is as follows:

**hbase> describe 'table name'**

**hbase(main):006:0> describe 'newtbl'**

**DESCRIPTION**

**ENABLED**