**Hbase Shell Commands**

**Start the Hbase Shell**

All following commands assume you are in hbase shell, which is started using following command:

hbase shell

You should see output similar to following:

15/10/15 12:30:52 WARN conf.Configuration: hadoop.native.lib is deprecated. Instead ,use io.native.lib.available  
HBase Shell; enter 'help<RETURN>' for list of supported commands.  
Type "exit<RETURN>" to leave the HBase Shell  
Version 0.92.1-cdh4.0.1, rUnknown, Thu Jun 28 18:13:01 PDT 2015

**Create a Table**

We can create a table using create command, we need to mention table name and column family name.

create ‘<table name>’, ’<column family>’

We will create a table named ‘Employee’ and column family ‘Personal’ and ‘Professional’.

Don’t use long column family names in production environment.

create 'Employee', 'Personal', ’Professional’

It will give following output:

0 row(s) in 1.1900 seconds

=> Hbase::Table - Employee

You can confirm if the table has been created using **list** command, to list all tables created in database.

hbase(main):002:0> list

TABLE

Employee

2 row(s) in 0.0450 seconds

**Disabling the Table**

We can delete a table and change its settings after disabling it using disable command. We can reenable it using enable command.

We can disable Employee table using following syntax:

disable <Table Name>

Example:

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

0 row(s) in 1.2485 seconds

**Is disabled**

We can use is\_disabled command to confirm if the specified table is disabled. Following is the syntax for the same:

hbase> is\_disabled 'table name'

It returns true if the table is disabled, otherwise it returns False

Example:

hbase(main):030:0> is\_disabled 'Employee'

true

0 row(s) in 0.0320 seconds

We can use command disable\_all to disable all tables that match regular expression.

hbase> disable\_all 'Emp.\*'

All tables starting with Emp would be disabled.

**Enabling the Table**

We can enable the table using following syntax:

enable 'Employee'

Example :

hbase(main):005:0> enable 'Employee'

0 row(s) in 0.3592 seconds

**Is Enabled**

We can use is\_enabled command to confirm if the specified table is enabled. Following is the syntax for the same:

hbase> is\_enabled 'table name'

It returns true if the table is enabled, otherwise it returns False

Example:

hbase(main):032:0> is\_ enabled 'Employee'

true

0 row(s) in 0.0450 seconds

**Describe**

To see the description of the table you can see describe command , following is the syntax for the same :

hbase> describe 'table name'

Example :

hbase(main):007:0> describe 'Employee'

**Alter**

Alter command is used to make changes to an existing table . You can change delete a column family to the table, maximum number of cells of a column family , set and delete table scope operators.

Alter Maximum No. of cells in a column family

Syntax:

hbase> alter <Table\_Name>, NAME ⇒ <Column\_family>, VERSIONS ⇒ <No.>

Example:

Here maximum no. of cells is 4.

hbase(main):003:0> alter 'Employee', NAME ⇒ 'Personal', VERSIONS ⇒ 4

Updating all regions with the new schema...

0/1 regions updated.

1/1 regions updated.

Done.

0 row(s) in 1.3045 seconds

**Table Scope Operators**

Using alter, we can remove and set table scope operators such as MEMSTORE\_FLUSHSIZE , DEFERRED\_LOG\_FLUSH, MAX\_FILESIZE and READONLY.

We can set table read only using following syntax:

hbase>alter <table\_name>, READONLY

Example:

hbase(main):006:0> alter 'Employee', READONLY

Updating all regions with the new schema...

0/1 regions updated.

1/1 regions updated.

Done.

0 row(s) in 1.5240 seconds

**Remove Table Scope Operators**

We can remove table scope operators using alter command . Given below is the syntax to remove ‘MAX\_FILESIZE’ from Employee table.

hbase> alter 'Employee', METHOD ⇒ 'table\_att\_unset', NAME ⇒ 'MAX\_FILESIZE'

**Delete Column Family**

We can delete a column family using alter command , following is the syntax for the same:

hbase> alter ‘table name’, ‘delete’ ⇒ ‘column family’

**Adding a New Column Family**

We can add a new column family, using below syntax:

alter 'tablename', NAME => 'newcolumnfamily', VERSIONS => 10

**Exists**

We can confirm the existence of the table using exists command.

hbase(main):028:0> exists 'Employee'

Table Employee does exist

0 row(s) in 0.0470 seconds

==================================================================

hbase(main):015:0> exists 'Demo'

Table Demo does not exist

0 row(s) in 0.0370 seconds

**Drop**

We can drop a table using drop command, before dropping a table we need to disable it.

hbase(main):012:0> disable 'Employee'

0 row(s) in 2.4544 seconds

hbase(main):013:0> drop 'Employee'

0 row(s) in 1.4560 seconds

hbase(main):014:0> exists 'Employee'

Table Employee does not exist

0 row(s) in 0.1680 seconds

Using drop\_all we can disable all tables matching regex using following syntax :

hbase> drop\_all ‘Emp.\*’

**Insert Data**

We can insert data using put command.

put ’<table name>’,’row\_key’,’<column\_family:column\_name>’,’<value>’

Example:

hbase(main):001:0> put 'Employee','1','Personal:Name','Ram'

0 row(s) in 0.3500 seconds

hbase(main):002:0> put 'Employee','1','Personal:City','Delhi'

0 row(s) in 0.1250 seconds

hbase(main):003:0> put 'Employee','1','Professional:Designation','Analyst'

0 row(s) in 0.2564 seconds

hbase(main):004:0> put 'Employee','1','Professional:Salary','12000'

0 row(s) in 0.0350 seconds

You can select records of the table using following command:

hbase(main):022:0> scan 'Employee'

**Read Data**

We can read data using get command.

Using following syntax:

get ’<table name>’,’<row\_key>’

Example:

hbase(main):014:0> get 'Employee', '1'

**Reading specific column**

Syntax:

hbase> get 'table name', ‘row\_key’, {COLUMN ⇒ ‘column\_family : column\_name ’}

Example:

hbase(main):021:0> get 'Employee', '1', {COLUMN ⇒ 'Personal:Name'}

**Delete Data**

We can delete data using delete command, using following syntax :

delete ‘<table name>’, ‘<row\_key>’, ‘<column\_name >’, ‘<time stamp>’

Example:

hbase(main):006:0> delete 'Employee', '1', 'Personal:Name',<time stamp>

Here you need to enter <time stamp> value as you read from your table , when you scan table.

Delete All

We can use ‘deleteall’ command , where we are deleting all cells of a row key.

hbase(main):004:0> deleteall 'Employee','1'

0 row(s) in 0.0420 seconds

**Count**

We can count the number of records in the table using count table.

Syntax:

count ‘<table\_name>’

Example :

hbase(main):025:0> count 'Employee'

**Truncate**

This table disables, drops and recreates existing table.

Syntax:

hbase> truncate 'table\_name'

Example :

hbase(main):014:0> truncate 'Employee'

**Exit Hbase Shell**

We can use exit command to exit from Hbase shell

hbase> exit