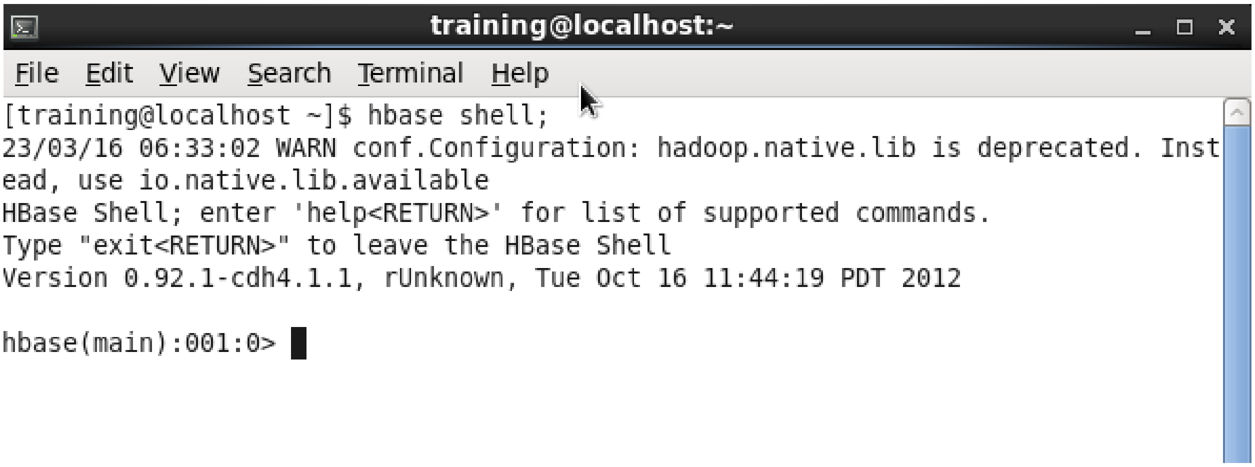
Name=Aditya Anal

HBase Tutorial:

1.hbase shell Command

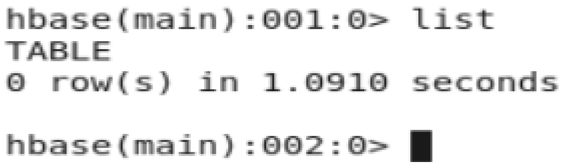
To start hbase



```
[training@localhost ~]$ hbase shell;
23/03/16 06:33:02 WARN conf.Configuration: hadoop.native.lib is deprecated. Inst
ead, 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-cdh4.1.1, rUnknown, Tue Oct 16 11:44:19 PDT 2012

hbase(main):001:0>
```

2.List Command

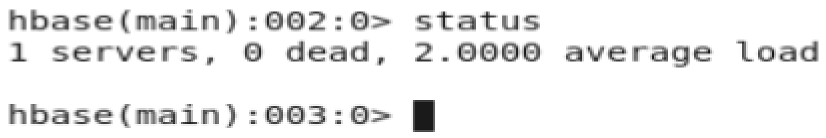
List is a command used to get the list of all the tables in HBase.



```
hbase(main):001:0> list
TABLE
0 row(s) in 1.0910 seconds

hbase(main):002:0>
```

3.Status Command

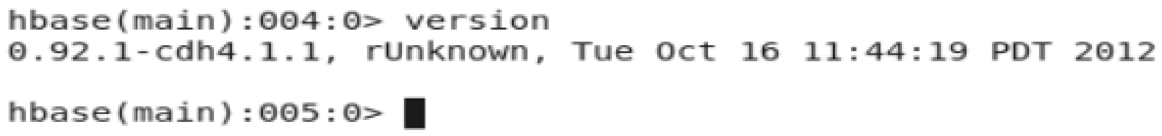
This command returns the status of the system including the details of the servers running on the system.



```
hbase(main):002:0> status
1 servers, 0 dead, 2.0000 average load

hbase(main):003:0> █
```

4. version Command:

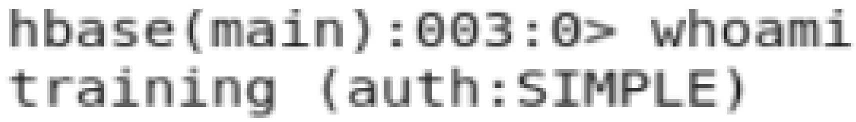
This command returns the version of HBase used in your system.



```
hbase(main):004:0> version
0.92.1-cdh4.1.1, rUnknown, Tue Oct 16 11:44:19 PDT 2012

hbase(main):005:0> █
```

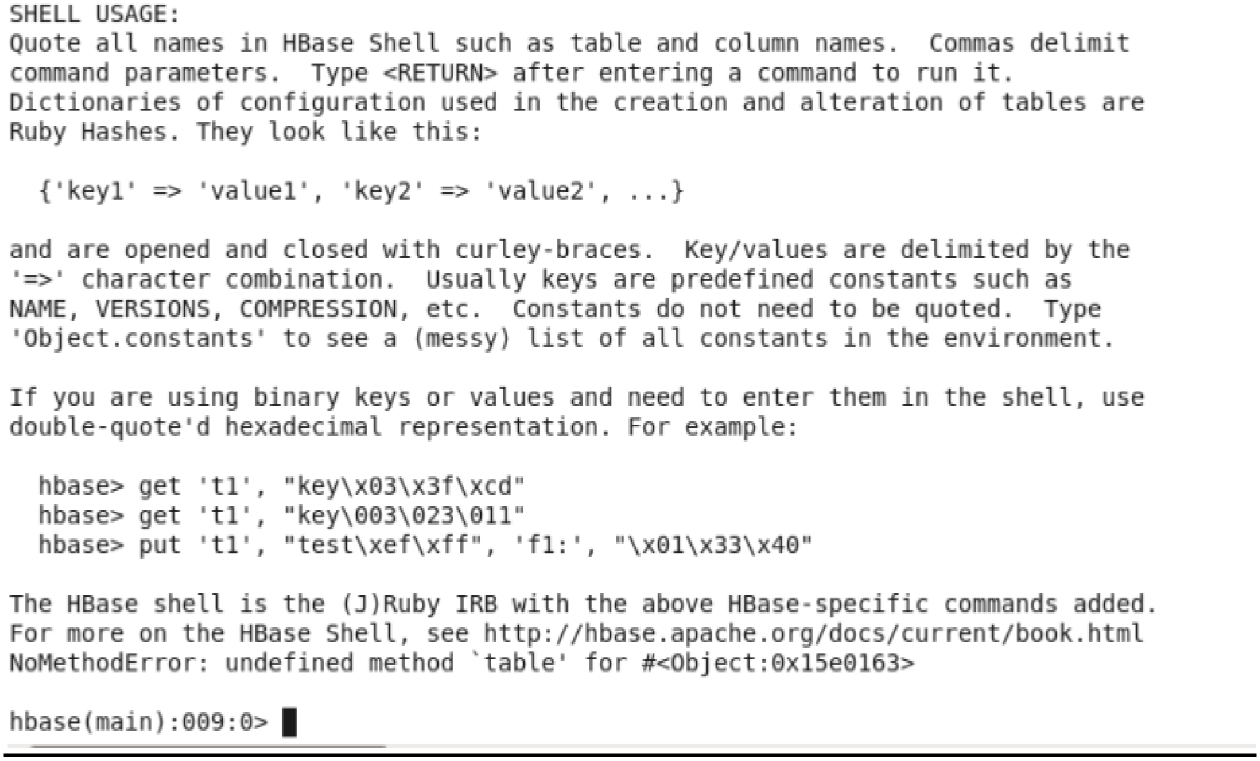
5. whoami Command

This command returns the user details of HBase.



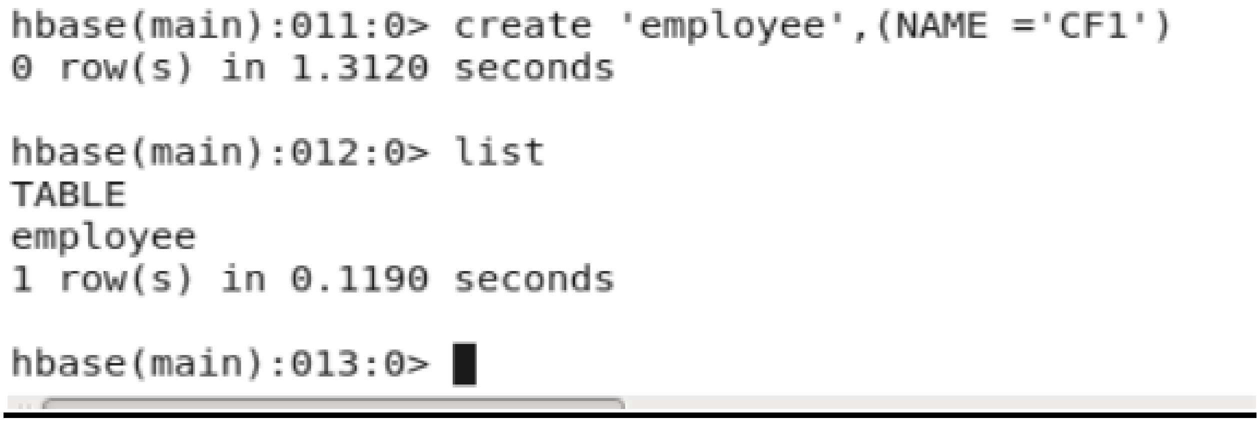
```
hbase(main):003:0> whoami
training (auth:SIMPLE)
```

6. table_help Command



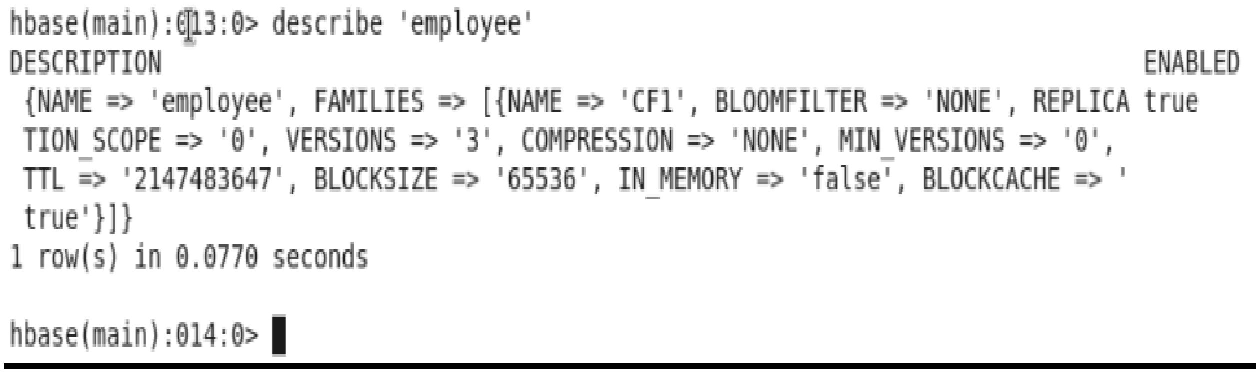
```
SHELL USAGE:
Quote all names in HBase Shell such as table and column names.  Commas delimit
command parameters.  Type <RETURN> after entering a command to run it.
Dictionaries of configuration used in the creation and alteration of tables are
Ruby Hashes. They look like this:

  {'key1' => 'value1', 'key2' => 'value2', ...}

and are opened and closed with curley-braces.  Key/values are delimited by the
'=>' character combination.  Usually keys are predefined constants such as
NAME, VERSIONS, COMPRESSION, etc.  Constants do not need to be quoted.  Type
'Object.constants' to see a (messy) list of all constants in the environment.

If you are using binary keys or values and need to enter them in the shell, use
double-quote'd hexadecimal representation. For example:

  hbase> get 't1', "key\x03\x3f\xcd"
  hbase> get 't1', "key\003\023\011"
  hbase> put 't1', "test\xef\xff", 'f1:', "\x01\x33\x40"

The HBase shell is the (J)Ruby IRB with the above HBase-specific commands added.
For more on the HBase Shell, see http://hbase.apache.org/docs/current/book.html
NoMethodError: undefined method `table' for #<Object:0x15e0163>

hbase(main):009:0>
```

7.Creating a Table using HBase Shell



```
hbase(main):011:0> create 'employee',(NAME ='CF1')
0 row(s) in 1.3120 seconds

hbase(main):012:0> list
TABLE
employee
1 row(s) in 0.1190 seconds

hbase(main):013:0> █
```

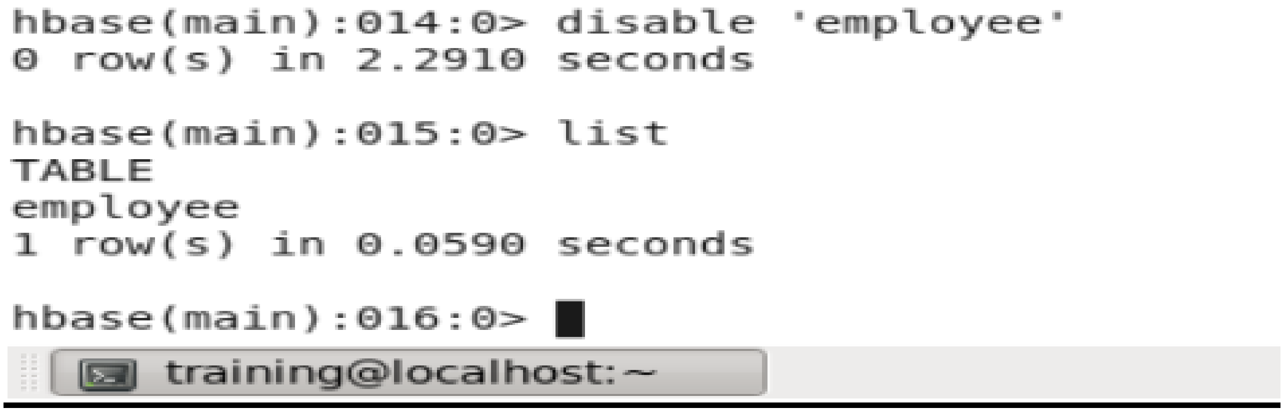
8.Describe Command



```
hbase(main):013:0> describe 'employee'
DESCRIPTION                                                                    ENABLED
 {NAME => 'employee', FAMILIES => [{NAME => 'CF1', BLOOMFILTER => 'NONE', REPLICA true
 TION_SCOPE => '0', VERSIONS => '3', COMPRESSION => 'NONE', MIN_VERSIONS => '0',
 TTL => '2147483647', BLOCKSIZE => '65536', IN_MEMORY => 'false', BLOCKCACHE => '
 true'}]}
1 row(s) in 0.0770 seconds

hbase(main):014:0> █
```

9.Disabling a Table using HBase Shell

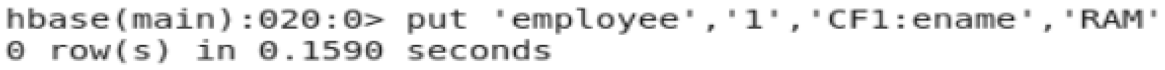
To delete a table or change its settings, you need to first disable the table using the disable command.



```
hbase(main):014:0> disable 'employee'
0 row(s) in 2.2910 seconds

hbase(main):015:0> list
TABLE
employee
1 row(s) in 0.0590 seconds

hbase(main):016:0> 
```

10.put Command

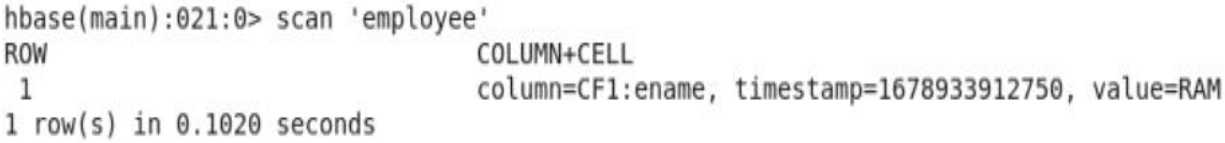
To insert data into table



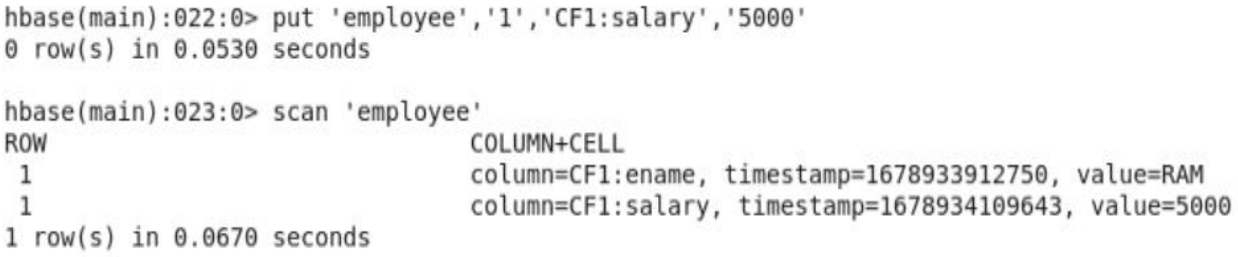
```
hbase(main):020:0> put 'employee','1','CF1:ename','RAM'
0 row(s) in 0.1590 seconds
```

11.scan Command

To retrieve the data.



```
hbase(main):021:0> scan 'employee'
ROW                              COLUMN+CELL
 1                               column=CF1:ename, timestamp=1678933912750, value=RAM
1 row(s) in 0.1020 seconds
```

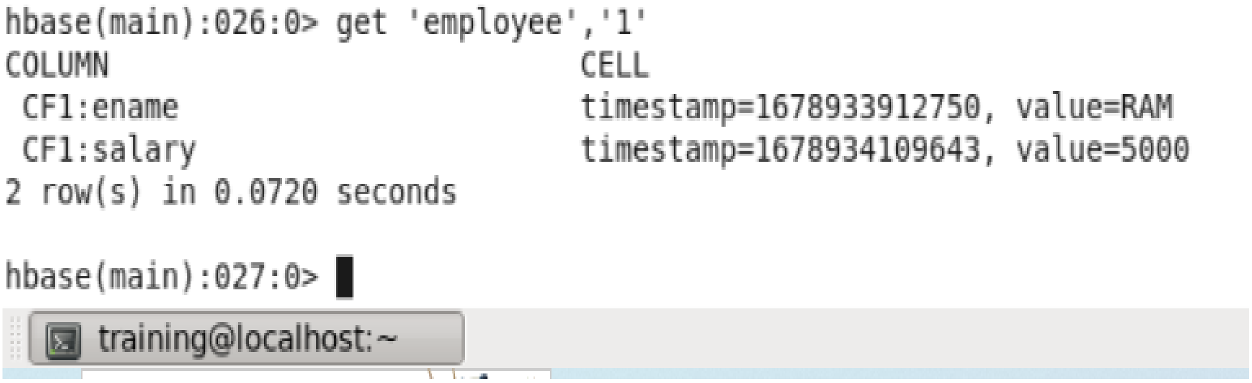


```
hbase(main):022:0> put 'employee','1','CF1:salary','5000'
0 row(s) in 0.0530 seconds

hbase(main):023:0> scan 'employee'
ROW                              COLUMN+CELL
 1                               column=CF1:ename, timestamp=1678933912750, value=RAM
 1                               column=CF1:salary, timestamp=1678934109643, value=5000
1 row(s) in 0.0670 seconds
```

12.get command

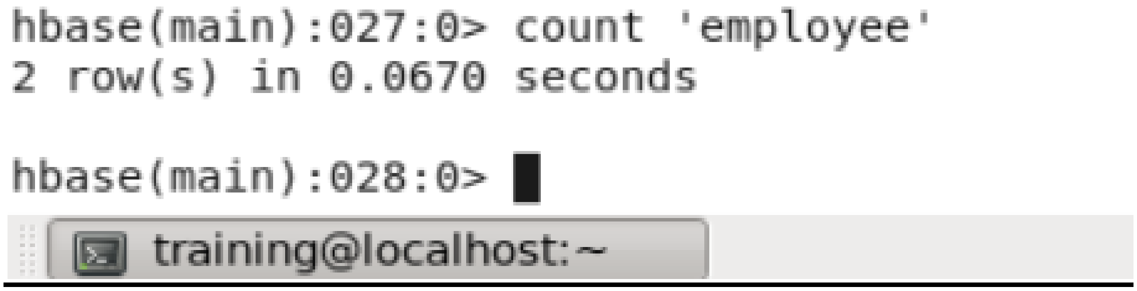
Get partiticular data from datset

Count



```
hbase(main):026:0> get 'employee','1'
COLUMN                           CELL
 CF1:ename                       timestamp=1678933912750, value=RAM
 CF1:salary                      timestamp=1678934109643, value=5000
2 row(s) in 0.0720 seconds

hbase(main):027:0> █
```

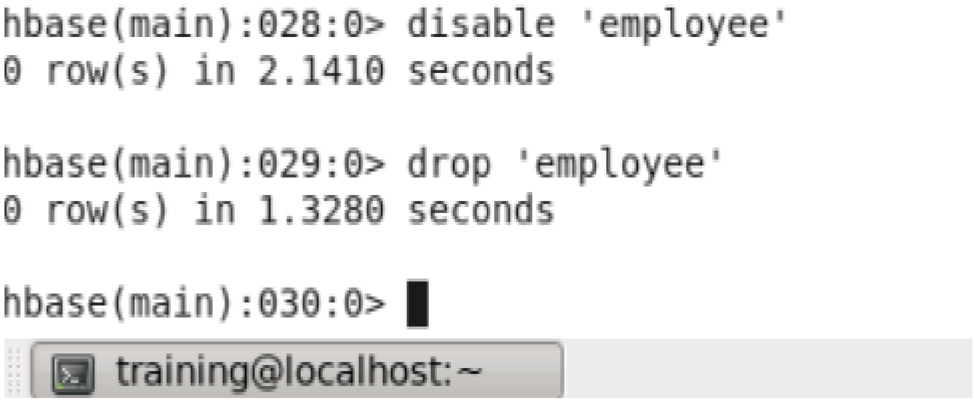
13.count Command



```
hbase(main):027:0> count 'employee'
2 row(s) in 0.0670 seconds

hbase(main):028:0>
```

14,Drop Command

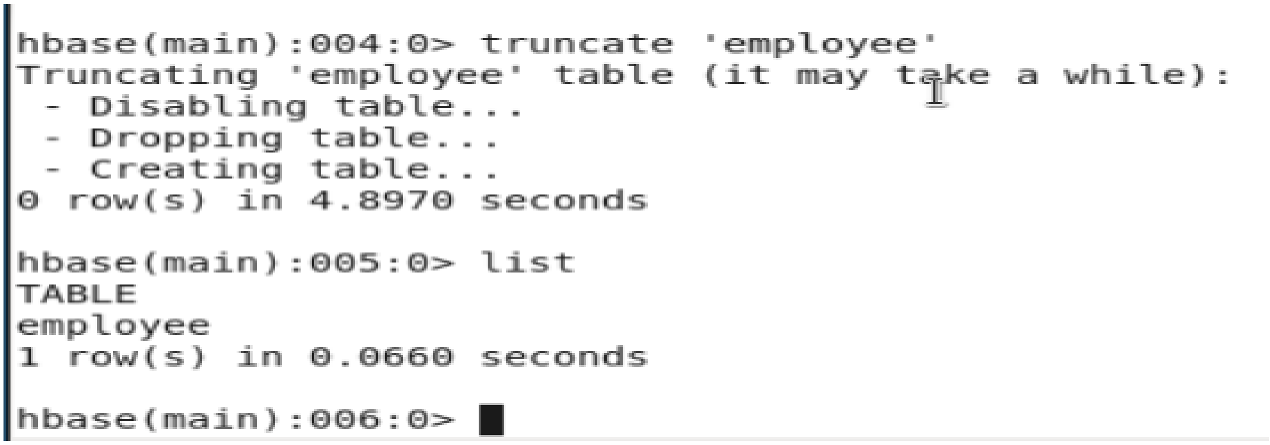
First do disable than deleted the table



```
hbase(main):028:0> disable 'employee'
0 row(s) in 2.1410 seconds

hbase(main):029:0> drop 'employee'
0 row(s) in 1.3280 seconds

hbase(main):030:0>
```

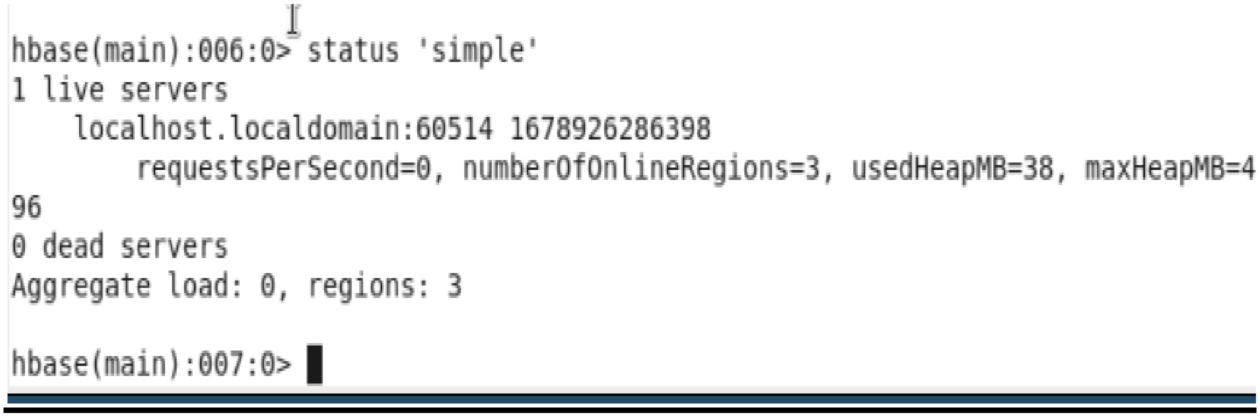
15.turncate command

To delete the data but not table



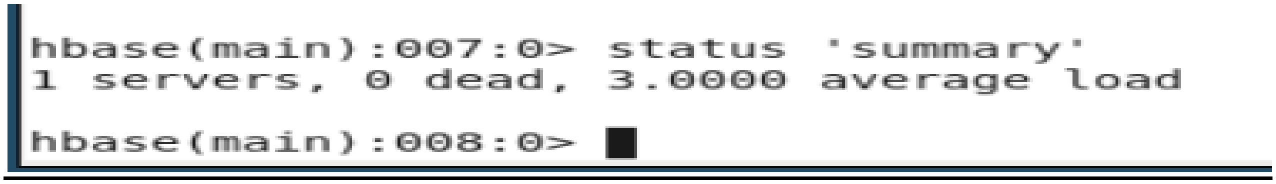
```
hbase(main):004:0> truncate 'employee'
Truncating 'employee' table (it may take a while):
 - Disabling table...
 - Dropping table...
 - Creating table...
0 row(s) in 4.8970 seconds

hbase(main):005:0> list
TABLE
employee
1 row(s) in 0.0660 seconds

hbase(main):006:0> 
```

16.status ‘sample’



```
hbase(main):006:0> status 'simple'
1 live servers
    localhost.localdomain:60514 1678926286398
        requestsPerSecond=0, numberOfOnlineRegions=3, usedHeapMB=38, maxHeapMB=4
96
0 dead servers
Aggregate load: 0, regions: 3

hbase(main):007:0> 
```

Status ‘summary’



```
hbase(main):007:0> status 'summary'
1 servers, 0 dead, 3.0000 average load

hbase(main):008:0> 
```