

## Lab Assignment 4

Submitted by: Aadarsha Chapagain

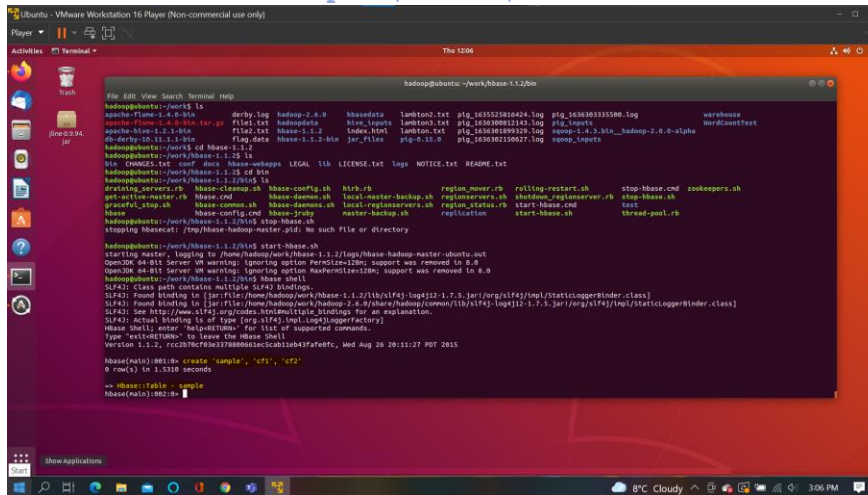
Execute below HBASE Commands and post the screenshots with one example each.

### 1. Create a Table in HBASE

Syntax:

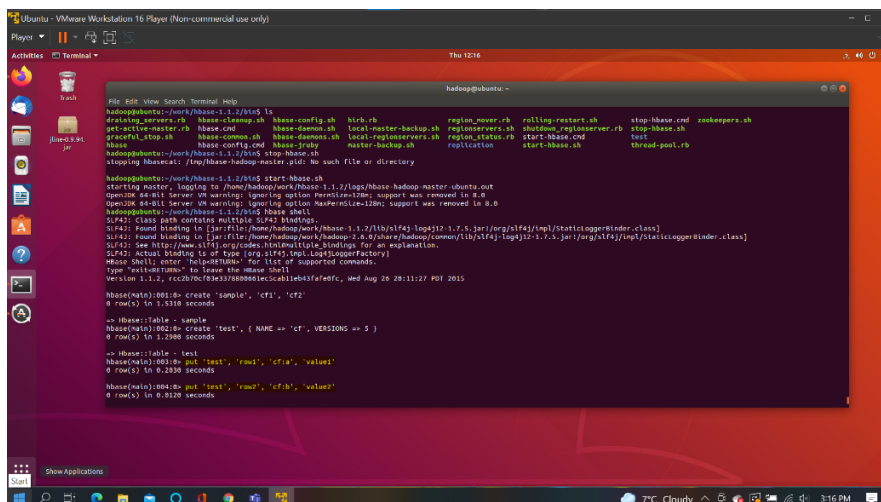
```
create 'tbl_name', 'column_family1', 'column_family2', ...
create 'tbl_name', { NAME => 'column_family1', ... }, { NAME =>
'column_family2', ... }, ...
```

Command: create 'sample', 'cf1', 'cf2'



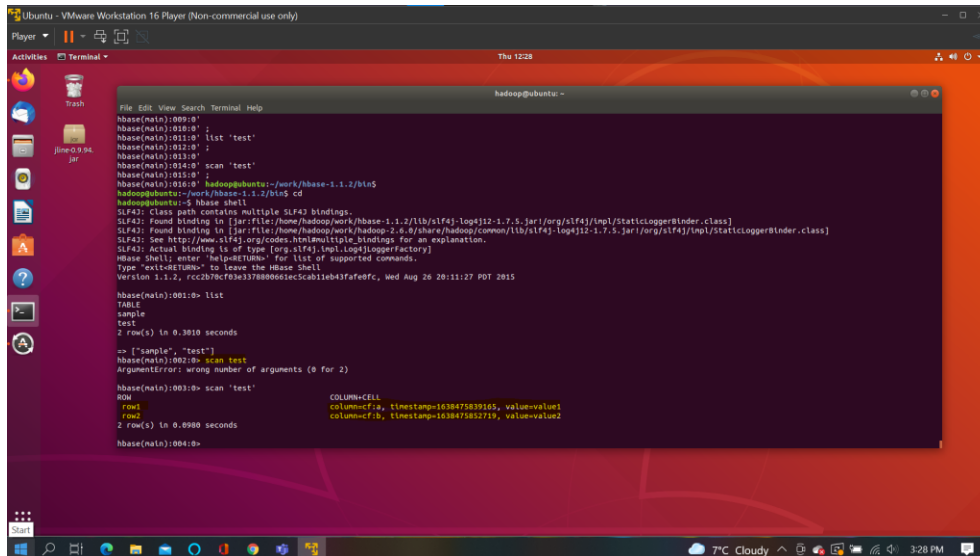
### 2. Insert/Update data in HBASE

```
put 'test', 'row1', 'cf:a', 'value1'
put 'test', 'row2', 'cf:b', 'value2'
```



### 3. Read Data from HBASE

#### scan test



```
hbase(main):009:0>
hbase(main):010:0>
hbase(main):011:0> list 'test'
hbase(main):012:0>
hbase(main):013:0>
hbase(main):014:0> scan 'test'
hbase(main):015:0>
hbase(main):016:0> hadoop@ubuntu:~/work/hbase-1.1.2/bin$ cd
hadoop@ubuntu:~$ hbase shell
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/hadoop/work/hbase-1.1.2/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/hadoop/work/hadoop-2.6.0/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple-bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
Hbase Shell; enter 'help@HBase' for list of supported commands.
Type 'exit@HBase' to leave the Hbase Shell
Version 1.1.2, rcc2b7cfc0e337880066sec5cab1eb43fafa0fc, Wed Aug 26 20:11:27 PDT 2015

hbase(main):001:0> list
TABLE
sample
test
2 row(s) in 0.3910 seconds

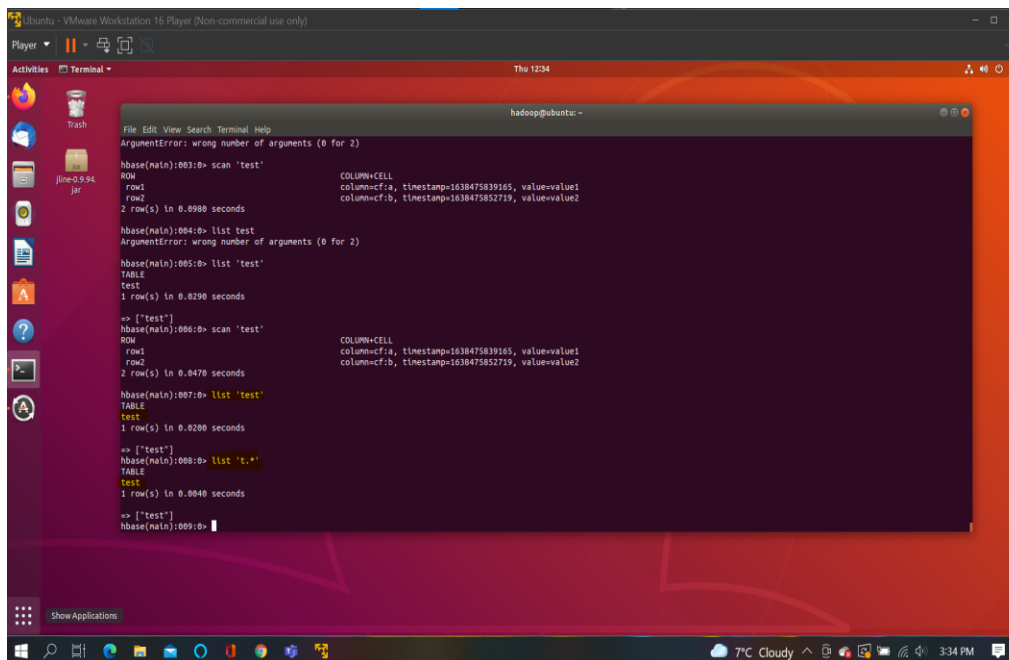
=> ["sample", "test"]
hbase(main):002:0> scan test
ArgumentError: wrong number of arguments (0 for 2)

hbase(main):003:0> scan 'test'
ROW                                COLUMN+CELL
row1                                column=cf:a, timestamp=1638475839165, value=value1
row2                                column=cf:b, timestamp=1638475852719, value=value2
2 row(s) in 0.0980 seconds

hbase(main):004:0>
```

### 4. List Data in HBASE

#### List <tablename>



```
ArgumentError: wrong number of arguments (0 for 2)

hbase(main):003:0> scan 'test'
ROW                                COLUMN+CELL
row1                                column=cf:a, timestamp=1638475839165, value=value1
row2                                column=cf:b, timestamp=1638475852719, value=value2
2 row(s) in 0.0980 seconds

hbase(main):004:0> list test
ArgumentError: wrong number of arguments (0 for 2)

hbase(main):005:0> list 'test'
TABLE
test
1 row(s) in 0.0290 seconds

=> ["test"]
hbase(main):006:0> scan 'test'
ROW                                COLUMN+CELL
row1                                column=cf:a, timestamp=1638475839165, value=value1
row2                                column=cf:b, timestamp=1638475852719, value=value2
2 row(s) in 0.0470 seconds

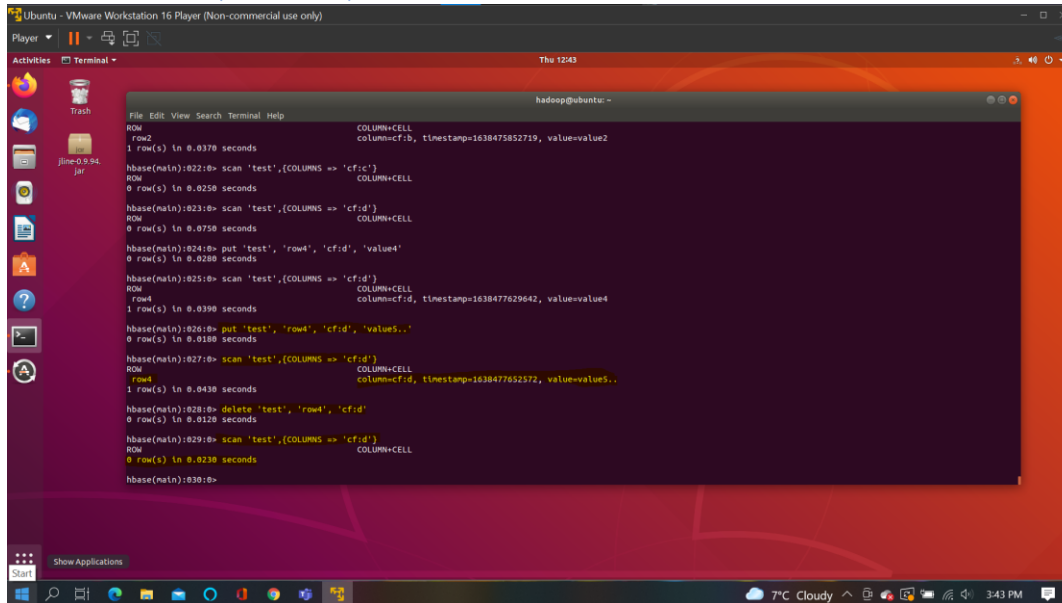
hbase(main):007:0> list "test"
TABLE
test
1 row(s) in 0.0200 seconds

=> ["test"]
hbase(main):008:0> list 't.'
TABLE
test
1 row(s) in 0.0040 seconds

=> ["test"]
hbase(main):009:0>
```

## 5. Delete Data from HBASE

```
delete 'tbl_name', 'row_key', 'column_family1:qualifier'  
delete 'test', 'row4', 'cf:d'
```

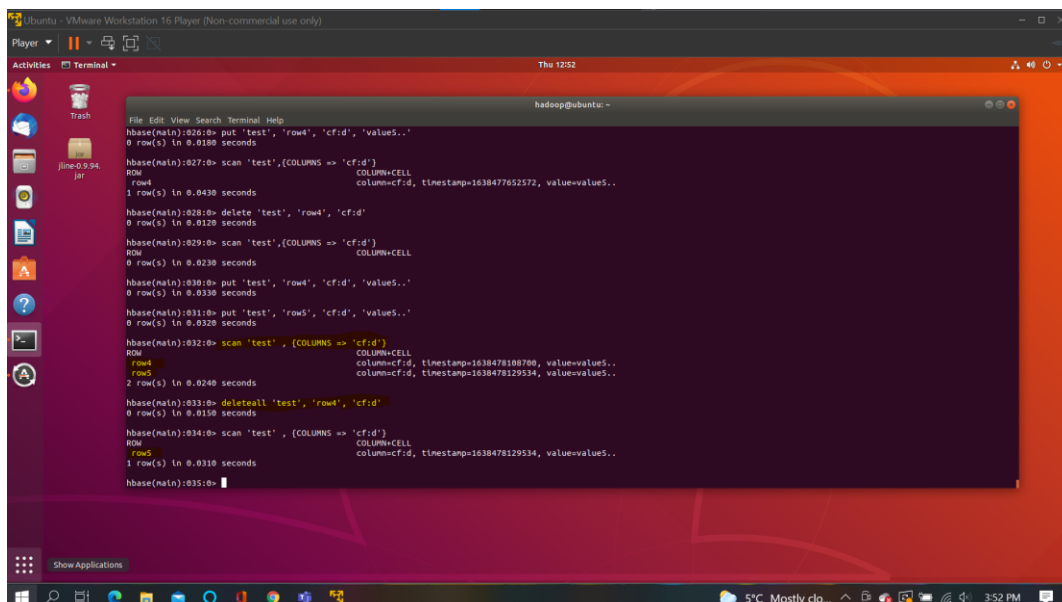


The screenshot shows a terminal window titled 'hadoop@ubuntu: ~' with the following commands and output:

```
File Edit View Search Terminal Help  
ROW  
Row2  
COLUMN+CELL  
column=cf:b, timestamp=1638475852719, value=value2  
1 row(s) in 0.0370 seconds  
hbase(main):022:0> scan 'test', [COLUMNS => 'cf:c']  
ROW  
COLUMN+CELL  
0 row(s) in 0.0250 seconds  
hbase(main):023:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
0 row(s) in 0.0750 seconds  
hbase(main):024:0> put 'test', 'row4', 'cf:d', 'value4'  
0 row(s) in 0.0280 seconds  
hbase(main):025:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
column=cf:d, timestamp=1638477629642, value=value4  
1 row(s) in 0.0390 seconds  
hbase(main):026:0> put 'test', 'row4', 'cf:d', 'value5..'  
0 row(s) in 0.0180 seconds  
hbase(main):027:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
column=cf:d, timestamp=1638477652572, value=value5..  
1 row(s) in 0.0430 seconds  
hbase(main):028:0> delete 'test', 'row4', 'cf:d'  
0 row(s) in 0.0120 seconds  
hbase(main):029:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
0 row(s) in 0.0230 seconds  
hbase(main):030:0>
```

## 6. Delete all Data from HBase

```
deleteall 'test', 'row4', 'cf:d'
```

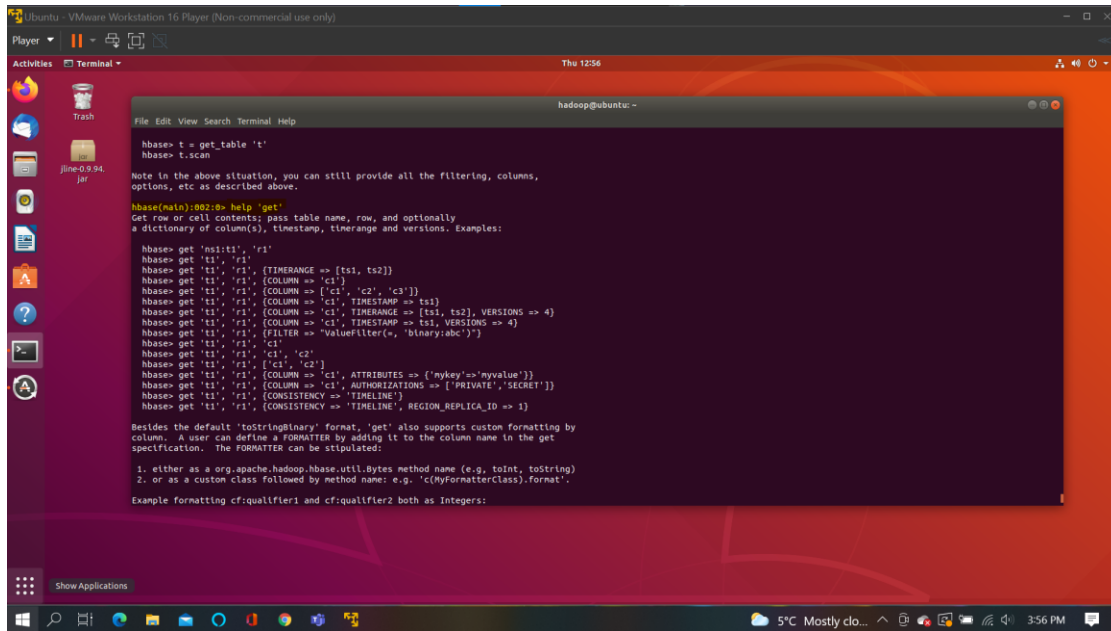


The screenshot shows a terminal window titled 'hadoop@ubuntu: ~' with the following commands and output:

```
File Edit View Search Terminal Help  
hbase(main):026:0> put 'test', 'row4', 'cf:d', 'value5..'  
0 row(s) in 0.0180 seconds  
hbase(main):027:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
column=cf:d, timestamp=1638477652572, value=value5..  
1 row(s) in 0.0430 seconds  
hbase(main):028:0> delete 'test', 'row4', 'cf:d'  
0 row(s) in 0.0120 seconds  
hbase(main):029:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
0 row(s) in 0.0230 seconds  
hbase(main):030:0> put 'test', 'row4', 'cf:d', 'value5..'  
0 row(s) in 0.0330 seconds  
hbase(main):031:0> put 'test', 'rows', 'cf:d', 'value5..'  
0 row(s) in 0.0320 seconds  
hbase(main):032:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
column=cf:d, timestamp=1638478100780, value=value5..  
2 row(s) in 0.0240 seconds  
hbase(main):033:0> deleteall 'test', 'row4', 'cf:d'  
0 row(s) in 0.0150 seconds  
hbase(main):034:0> scan 'test', [COLUMNS => 'cf:d']  
ROW  
COLUMN+CELL  
column=cf:d, timestamp=1638478129534, value=value5..  
1 row(s) in 0.0310 seconds  
hbase(main):035:0>
```

## 7. Help Commands in HBase

### Help 'get'



```
hbase> t = get_table 't'
hbase> t.scan

Note in the above situation, you can still provide all the filtering, columns,
options, etc as described above.

hbase(main):002:0> help 'get'
Get row or cell contents; pass table name, row, and optionally
a dictionary of column(s), timestamp, timerange and versions. Examples:

hbase> get 'ns1:t1', 'r1'
hbase> get 't1', 'r1'
hbase> get 't1', 'r1', {TIMERANGE => [ts1, ts2]}
hbase> get 't1', 'r1', {COLUMN => 'c1'}
hbase> get 't1', 'r1', {COLUMN => ['c1', 'c2', 'c3']}
hbase> get 't1', 'r1', {COLUMN => 'c1', TIMESTAMP => ts1}
hbase> get 't1', 'r1', {COLUMN => 'c1', TIMERANGE => [ts1, ts2], VERSIONS => 4}
hbase> get 't1', 'r1', {COLUMN => 'c1', TIMESTAMP => ts1, VERSIONS => 4}
hbase> get 't1', 'r1', {FILTER => 'ValueFilter(, 'binary:abc')'}
hbase> get 't1', 'r1', 'c1'
hbase> get 't1', 'r1', 'c1', 'c2'
hbase> get 't1', 'r1', {'c1', 'c2'}
hbase> get 't1', 'r1', {COLUMN => 'c1', ATTRIBUTES => {'mykey'=>'myvalue'}}
hbase> get 't1', 'r1', {COLUMN => 'c1', AUTHORIZATIONS => ['PRIVATE', 'SECRET']}
hbase> get 't1', 'r1', {CONSISTENCY => 'TIMELINE'}
hbase> get 't1', 'r1', {CONSISTENCY => 'TIMELINE', REGION_REPLICA_ID => 1}

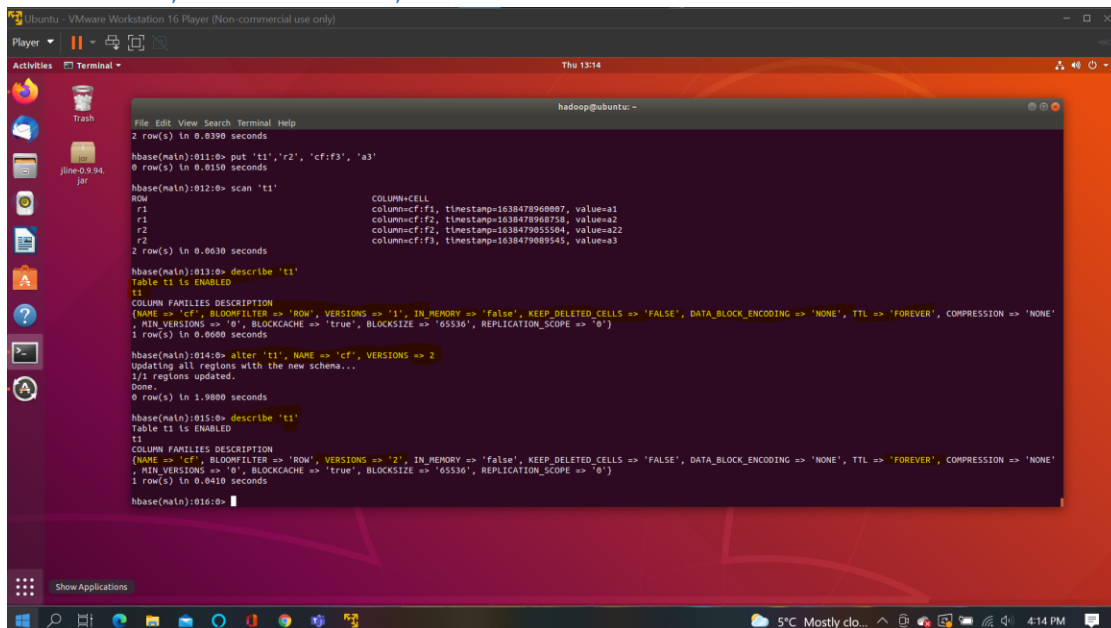
Besides the default 'toStringBinary' format, 'get' also supports custom formatting by
column. A user can define a FORMATTER by adding it to the column name in the get
specification. The FORMATTER can be stipulated:

1. either as a org.apache.hadoop.hbase.util.Bytes method name (e.g. toInt, toString)
2. or as a custom class followed by method name: e.g. 'c(myformatterClass).format'

Example formatting cf:qualifier1 and cf:qualifier2 both as integers:
```

## 8. Alter a table in HBase

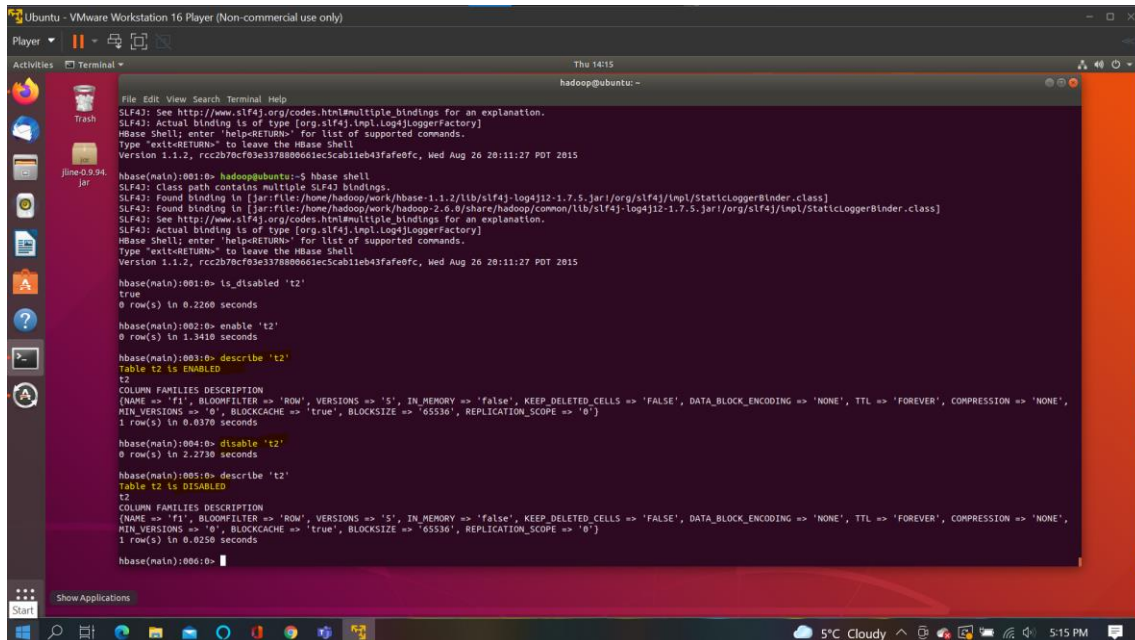
alter 't1', NAME => 'cf', VERSIONS => 2



```
2 row(s) in 0.0390 seconds
hbase(main):011:0> put 't1', 'r2', 'cf:f3', 'a3'
0 row(s) in 0.0150 seconds
hbase(main):012:0> scan 't1'
ROW COLUMN+CELL
r1 column=cf:f1, timestamp=1638478600007, value=a1
r2 column=cf:f2, timestamp=1638478608750, value=a2
r2 column=cf:f3, timestamp=1638479055504, value=a3
2 row(s) in 0.0630 seconds
hbase(main):013:0> describe 't1'
Table t1 is ENABLED
COLUMN FAMILIES DESCRIPTION
(NAME => 'cf', BLOOMFILTER => 'ROW', VERSIONS => '1', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0600 seconds
hbase(main):014:0> alter 't1', NAME => 'cf', VERSIONS => 2
Updating all regions with the new schema...
1/1 regions updated.
Done.
0 row(s) in 1.9800 seconds
hbase(main):015:0> describe 't1'
Table t1 is ENABLED
COLUMN FAMILIES DESCRIPTION
(NAME => 'cf', BLOOMFILTER => 'ROW', VERSIONS => '2', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0410 seconds
hbase(main):016:0> |
```

## 9. Disable a table in HBase

### Disable 't2'



```
file Edit View Search Terminal Help
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
HBase Shell; enter 'help=RETURN' for list of supported commands.
Type "exit=RETURN" to leave the HBase Shell
Version 1.1.2, rcc2b7cf03e337880661ec5cab1eb43fafa0fc, Wed Aug 26 20:11:27 PDT 2015

hbase(main):001:0> hbase shell
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/hadoop/work/hbase-1.1.2/lib/slf4j-log4j12-1.7.5.jar/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/hadoop/work/hadoop-2.6.0/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
HBase Shell; enter 'help=RETURN' for list of supported commands.
Type "exit=RETURN" to leave the HBase Shell
Version 1.1.2, rcc2b7cf03e337880661ec5cab1eb43fafa0fc, Wed Aug 26 20:11:27 PDT 2015

hbase(main):001:0> is_disabled 't2'
true
0 row(s) in 0.2260 seconds

hbase(main):002:0> enable 't2'
0 row(s) in 1.3410 seconds

hbase(main):003:0> describe 't2'
Table t2 is ENABLED
t2
COLUMN FAMILIES DESCRIPTION
(NAME => 'f1', BLOOMFILTER => 'ROW', VERSIONS => '5', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0370 seconds

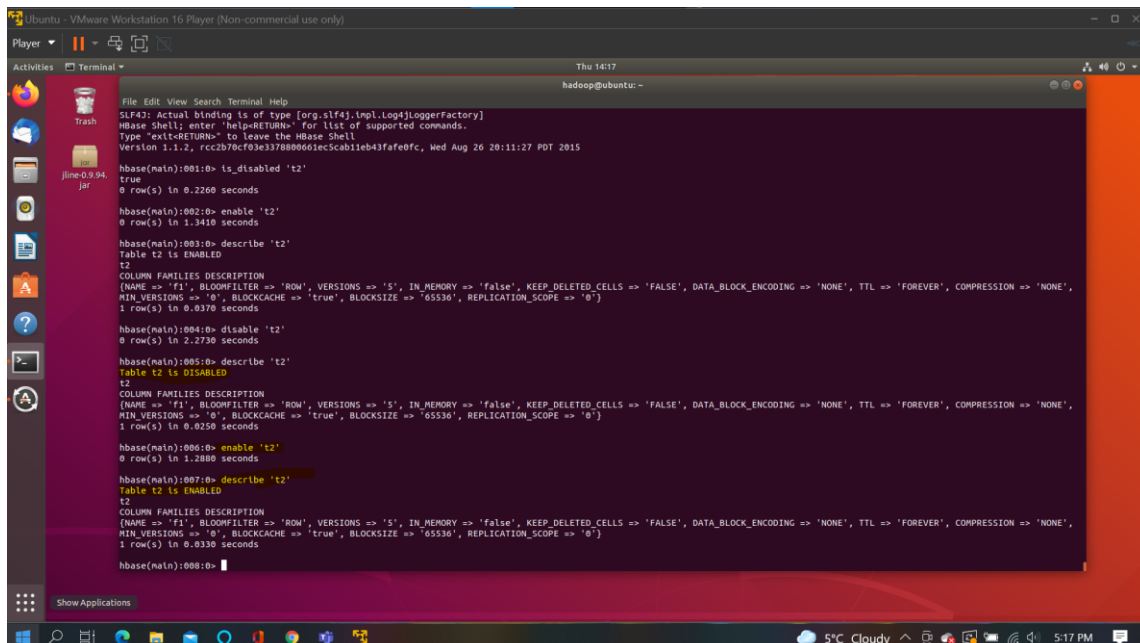
hbase(main):004:0> disable 't2'
0 row(s) in 2.2730 seconds

hbase(main):005:0> describe 't2'
Table t2 is DISABLED
t2
COLUMN FAMILIES DESCRIPTION
(NAME => 'f1', BLOOMFILTER => 'ROW', VERSIONS => '5', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0250 seconds

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

## 10. Enable a table in HBase

### Enable 't2'



```
file Edit View Search Terminal Help
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
HBase Shell; enter 'help=RETURN' for list of supported commands.
Type "exit=RETURN" to leave the HBase Shell
Version 1.1.2, rcc2b7cf03e337880661ec5cab1eb43fafa0fc, Wed Aug 26 20:11:27 PDT 2015

hbase(main):001:0> is_disabled 't2'
true
0 row(s) in 0.2260 seconds

hbase(main):002:0> enable 't2'
0 row(s) in 1.3410 seconds

hbase(main):003:0> describe 't2'
Table t2 is ENABLED
t2
COLUMN FAMILIES DESCRIPTION
(NAME => 'f1', BLOOMFILTER => 'ROW', VERSIONS => '5', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0370 seconds

hbase(main):004:0> disable 't2'
0 row(s) in 2.2730 seconds

hbase(main):005:0> describe 't2'
Table t2 is DISABLED
t2
COLUMN FAMILIES DESCRIPTION
(NAME => 'f1', BLOOMFILTER => 'ROW', VERSIONS => '5', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0250 seconds

hbase(main):006:0> enable 't2'
0 row(s) in 1.2880 seconds

hbase(main):007:0> describe 't2'
Table t2 is ENABLED
t2
COLUMN FAMILIES DESCRIPTION
(NAME => 'f1', BLOOMFILTER => 'ROW', VERSIONS => '5', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE',
MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0')
1 row(s) in 0.0330 seconds

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